PI JDBC 2012

Administrator Guide

OSIsoft, LLC

777 Davis St., Suite 250
San Leandro, CA 94577 USA

Tel: (01) 510-297-5800
Fax: (01) 510-357-8136
Web: http://www.osisoft.com

OSIsoft Australia • Perth, Australia

OSIsoft Europe GmbH • Frankfurt, Germany

OSIsoft Asia Pte Ltd. • Singapore

OSIsoft Canada ULC • Montreal & Calgary, Canada

OSIsoft, LLC Representative Office • Shanghai, People's Republic of China

OSIsoft Japan KK • Tokyo, Japan

OSIsoft Mexico S. De R.L. De C.V. • Mexico City, Mexico

OSIsoft do Brasil Sistemas Ltda. • Sao Paulo, Brazil

OSIsoft France EURL • Paris, France

PI JDBC 2012 Administrator Guide

Version: 1.3.0.0251

Published: 3/21/2013

Table of Contents

Chapter 1 Introduction 1

In this Guide 1
About the OSIsoft PI Data Access Suite 2
Architecture 2
Deployment Options 3

Installation 5

Installation Components 5
System Requirements 5
Install PI SQL Data Access Server 6
Install PI JDBC Driver on Windows 12
Install PI JDBC Driver on Linux 17

Using PI JDBC 21

Configuration 21
Functionality 24
Use Cases 26
PI JDBC Driver Limitations 29

Troubleshooting 31

Run PI SQL DAS Interactively 31
Connection Messages and Errors 31
Verify Java Version 34
OpenSSL Library 34

Tested PI JDBC Scenarios 39

Technical Support and Resources 41

Index 43

Chapter 1

Introduction

PI JDBC Driver is a JDBC 4.0 compliant driver that provides robust data access to the PI System through SQL queries. PI JDBC Driver offers much of the same functionality as the PI OLEDB Enterprise Provider, and is also backward compatible with the behavior of the classic PI OLEDB Provider.

Java is a programming language that allows you to create platform independent software. This platform independence is made possible by the Java Runtime Environment (JRE), which handles Java code and makes it available to run on most operating systems. Typically, you would select a Java programming environment when there is a demand to run software in both Linux and Windows.

Java Database Connectivity (JDBC) is an API that is built into every JRE and defines how a client can access a database. For relational database access, Java applications should use JDBC.

This version of PI JDBC is intended for use with Java applications and middleware, but is not meant for installation on many end-user computers. PI OLEDB and PI OLEDB Enterprise, although multithreaded, were not designed as server back ends and will only perform with a limited amount of parallel connections.

Later versions of PI JDBC will allow for scaling up to many concurrent user connections. PI JDBC will then migrate to a type 3 JDBC driver. See *PI JDBC Driver Limitations* (page 29) for information about other limitations.

PI JDBC Driver is a member of the *PI Data Access product suite* (page 2).

In this Guide

This guide includes procedures to install and configure PI JDBC Driver and PI SQL Data Access Server on Windows and Linux operating systems. It also provides information about how to use and troubleshoot PI JDBC Driver.

Users of this guide should be familiar with:

- Java
- JDBC technologies
- PI OLEDB and PI OLEDB Enterprise

About the OSIsoft PI Data Access Suite

The OSIsoft PI Data Access product suite is designed to support implementation of custom applications on top of the PI System, as well as integration of PI System data with other applications and business systems such as Microsoft Office or SQL Server, Enterprise Resource Planning systems (ERPs), Web portals, and maintenance systems, just to name a few.

The PI Data Access suite of products covers a wide range of use cases in various environments, programming languages, operating systems and infrastructures. Products include:

- SQL-based data access (PI OLEDB Provider, PI OLEDB Enterprise, PI JDBC Driver)
- OPC specifications (PI OPC DA/HDA Server)
- Service-oriented architecture (PI Web Services)
- Programmatic access (PI SDK and AF SDK)

Licensing for the PI Data Access products is divided into development and runtime licenses. Developers and integrators obtain development licenses for most PI Data Access components through their individual membership to the OSIsoft Virtual Campus (*vCampus* (*http://vCampus.osisoft.com*)) program. For details, see the OSIsoft vCampus *Frequently Asked Questions http://vCampus.osisoft.com/content/FAQ.aspx*.

The PI System Access (PSA) license enables end users to access PI System data, including time-series data in PI Servers and asset metadata in AF Servers. PSA is a runtime license to access PI System data using any of the programmatic access methods licensed through the PSA, including PI Web Services. For more information, see the *OSIsoft Web site* (*http://www.osisoft.com*) or contact *OSIsoft Technical Support* (page 41).

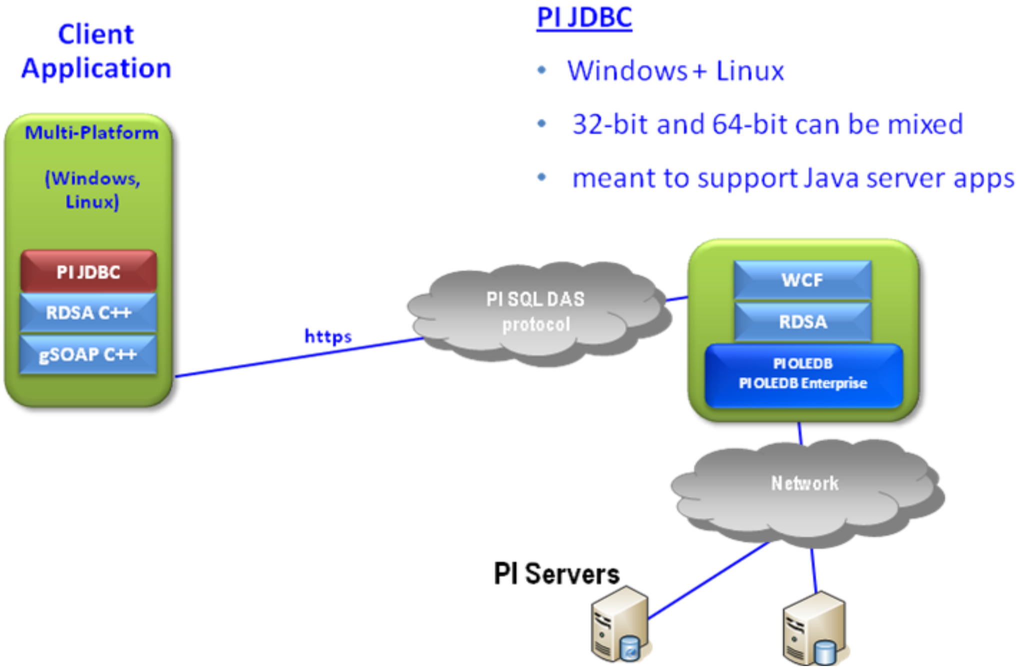
Architecture

The PI JDBC driver is an implementation of the JDBC standard based on the JDBC 4.0 API (Java Platform SE 6). PI JDBC is a type 1 JDBC driver (bridge) that employs the PI OLEDB Enterprise and PI OLEDB query engines to connect to the PI System and execute queries.

Note: For more information, see *Types of JDBC technology drivers* (*http://www.oracle.com/technetwork/java/overview-141217.html*) on the Oracle Technology Network Web site.

Communication from PI JDBC to the PI System requires using the PI SQL Data Access Server (PI SQL DAS). PI SQL DAS serves as a gateway between PI JDBC and PI OLEDB Enterprise/PI OLEDB. It provides secure network communication through HTTPS to PI JDBC and executes queries as an OLE DB client, also known as the consumer.

PI OLEDB Enterprise/PI OLEDB and PI SQL DAS run on Windows whereas PI JDBC is supported on Windows and Linux operating systems.



Deployment Options

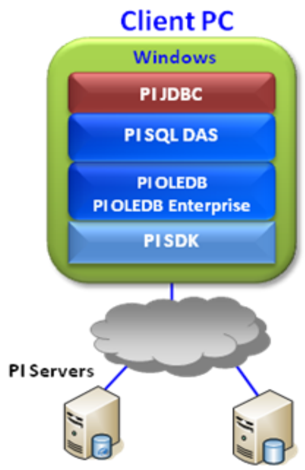
PI JDBC can be deployed in various combinations. PI JDBC and PI SQL DAS can run on different architectures (32-bit or 64-bit). In addition, PI JDBC can run on Linux or Windows. For combinations tested, see *Tested PI JDBC Scenarios* (page 39).

The resulting combinations can be categorized as *standalone* (page 4) and *middleware* (page 4) scenarios.

Note: For best performance, OSIsoft recommends that you use 64-bit operating systems whenever possible. The PI SQL DAS will especially benefit from running on a 64-bit Windows.

Multiple standalone and middleware configurations can be used as needed. This is useful if performance of one application should not be influenced by queries of another one.

Standalone Deployment

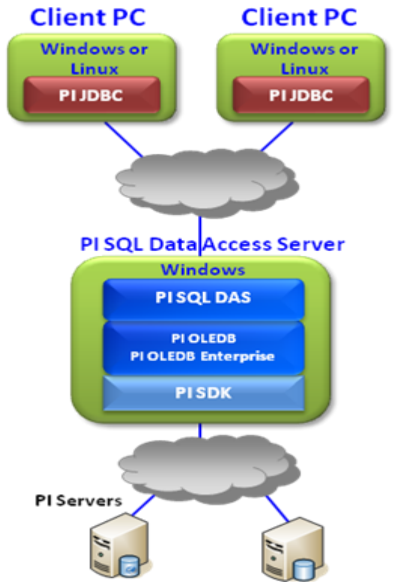


Notes

- All products installed on one machine
- Windows only
- Supports standalone Java applications

Use the standalone deployment if a single Java application needs to be supported with maximum performance.

Middleware Deployment



Notes

- PI JDBC on clients, other products on server
- Windows or Linux
- Support for Java server applications, limited number of clients

Middleware deployment takes advantage cross-platform support, for example, when supporting Java applications on Linux.

Chapter 2

Installation

Installation Components

The PI JDBC setup kits are distributed as four self-extracting executables and one compressed tar archive. PI JDBC consists of these components:

- PI SQL Data Access Server is the gateway between PI JDBC driver and PI OLEDB Enterprise/PI OLEDB providers.
- PI JDBC driver on Windows is the JDBC client for Windows operating systems.
- PI JDBC driver on Linux is the JDBC client for Linux operating systems.

Note: The setup kits for each of these components include both 32- and 64-bit versions.

System Requirements

PI SQL DAS requires:

- Windows XP or later
- .NET 4.0
- PI OLEDB
- PI OLEDB Enterprise

PI JDBC driver on Windows requires:

- Windows XP or later
- Java Runtime (JRE) 1.6.0.0 or higher

PI JDBC driver on Linux requires:

- Java Runtime (JRE) 1.6.0.0 or higher
- OpenSSL 0.9.8 or OpenSSL 1.0.0

Further dependencies are based on the OpenSSL version being used:

- OpenSSL 0.9.8
  - `libstdc++ 4.2`, including GLIBCXX_3.4.9 (standard GNU C++ shared library)
  - `glibc 2.4` (standard GNU C shared library)
  - `libgcc 4.1.2` (standard GNU C Compiler runtime library)

- OpenSSL 1.0.0
  - `glibc 2.11`

Note: The required Linux libraries are optional components and may need to be enabled or updated depending on the Linux distribution you are using. For additional information, see *OpenSSL Library* (page 34).

For further details about the PI JDBC release you are using, see the PI JDBC 2012 Release Notes.

Install PI SQL Data Access Server

To install PI SQL Data Access Server:

1. Select either the *standalone* (page 4) or *middleware* (page 4) deployment option.
2. Verify whether the computer you will use meets the *System Requirements* (page 5).
3. Install the *OSIsoft Prerequisites Kit* (*http://techsupport.osisoft.com/Products/Prerequisite+Kits/Prerequisite+Kits+Overview.htm*), available at the OSIsoft Technical Support Web site.
4. Run the *PI SQL DAS setup kit* (page 6).

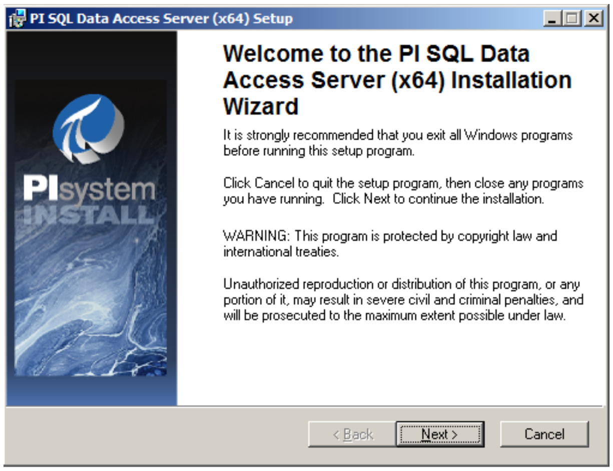
Run the PI SQL DAS Setup Kit

Note: The installation must be run from an account that has administrative privileges.

Run the setup kit: `pisqldas_1.`3.0.*xxxx*`.exe`

where

- *xxxx* is the build number



Welcome to the PI SQL Data Access Server (x64) Installation Wizard

It is strongly recommended that you exit all Windows programs before running this setup program.

Click Cancel to quit the setup program, then close any programs you have running. Click Next to continue the installation.

WARNING: This program is protected by copyright law and international treaties.

Unauthorized reproduction or distribution of this program, or any portion of it, may result in severe civil and criminal penalties, and will be prosecuted to the maximum extent possible under law.

Note: The installation of 32-bit and 64-bit versions of PI SQL DAS on the same machine is not supported. When you run the installation on a 64-bit Windows operating system, the 64-bit version of PI SQL DAS is automatically selected.

Silent Installation

The PI SQL DAS setup kit extracts several installation modules. The components of the installation process, their order, and the arguments used to launch the components are provided in a configuration file named `setup.ini`. If you modify this file, you can provide different command line arguments for different stages of the setup. This may be useful within a well-controlled environment with options that are known in advance, such as in the case of an embedded installation. The setup kit also contains a file named `silent.ini` that contains modifications to `setup.ini` that are typically needed to run a silent installation. You can augment these arguments by adding any of the options described below.

Individual arguments must contain no spaces unless they are surrounded by quotes.

Argument	Description
`/i`	Specifies an installation
`/qn`	Specifies the "quiet mode" and suppresses dialog boxes and prompts
`ALLUSERS`	Specifies the per-machine or per-user installation context. Use a value of **1** for silent installations.
`REBOOT`	Restarts the computer. Use a value of **Suppress** for silent installations.

Use this syntax for a silent installation of a single component:

```
msiexec.exe /i pisqldas_<version>-x86.msi REBOOT=Suppress
ALLUSERS=1 /qn
msiexec.exe /i pisqldas_<version>-x64.msi REBOOT=Suppress
ALLUSERS=1 /qn
```

Replace `<version>` with the current version string.

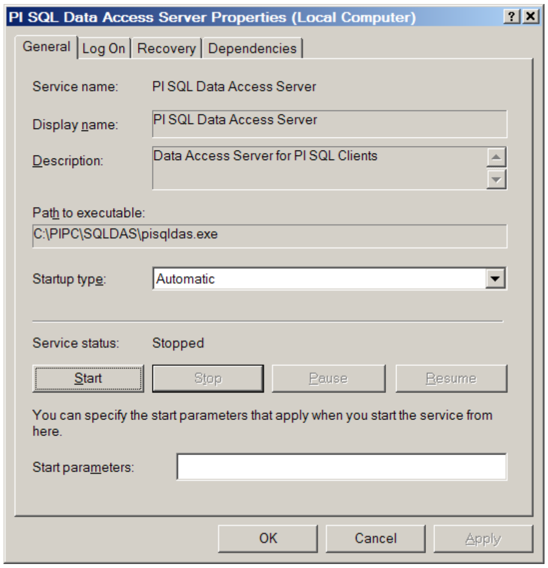
Note: To run the complete package in silent mode, replace the setup.ini file with silent.ini and run Setup.exe.

Files

The list of files installed is provided in the release notes document `PI JDBC 2012 Release Notes.pdf`.

Installation Result

Files are installed in the `[PIHOME]\SQLDAS` directory. The Windows Service **PI SQL Data Access Server** gets registered with startup type **Automatic**, and is started.



PI SQL Data Access Server Properties (Local Computer)

Service name: PI SQL Data Access Server

Display name: PI SQL Data Access Server

Description: Data Access Server for PI SQL Clients

Path to executable:
C:\PIPC\SQLDAS\pisqldas.exe

Startup type: Automatic

Service status: Stopped

You can specify the start parameters that apply when you start the service from here.

Start parameters:

The PI SQL Data Access Server is automatically configured to use a SELF-SIGNED certificate, bound to port **5461**. If your IT policies allow working with SELF-SIGNED certificates then the installation procedure is complete and you can skip the optional *configuration of PI SQL Data Access Server* (page 8).

Configure PI SQL Data Access Server (optional)

PI SQL Data Access Server is a self-hosted Windows Communication Foundation (WCF) service that uses HTTP transport and SSL security. As a result, a port is bound with an X.509 certificate.

Secure Sockets Layer (SSL) uses certificates on the client and server to store encryption keys. The server provides its SSL certificate when a connection is made so that the client can verify the identity of the server. The server can also request a certificate from the client to provide mutual authentication of both sides of the connection, but this mechanism is currently not used for PI SQL DAS.

Certificates are stored in a centralized store according to the IP address and port number of the connection. The special IP address **0.0.0.0** matches any IP address for the local machine. You must have administrative privileges to modify the certificates stored on the computer.

If you are required to use enterprise-type security certificates, use the information here to configure an enterprise certificate for PI SQL DAS and bind it to the port used by PI SQL DAS.

Configuration Tool

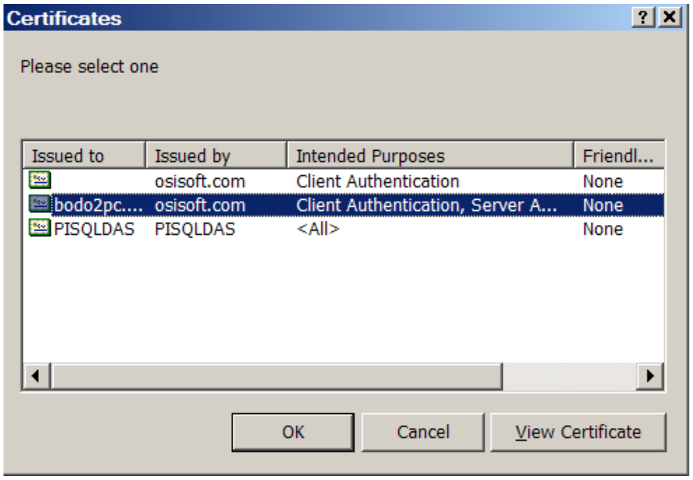
Use the **pisqldasAutoConfig.exe** configuration tool for all basic configurations of PI SQL DAS. The tool is located in the `[PIHOME]\SQLDAS` directory, where `[PIHOME]` is your `PIPC` installation directory.

If your IT policies require that you use enterprise type certificates, you can use the tool to configure an enterprise certificate for PI SQL DAS and bind it to the port used by PI SQL DAS.

1. In a command prompt, run the tool with parameter **-e**. For example:

   ```
   C:\PIPC\SQLDAS>pisqldasAutoConfig.exe -e
   ```

2. Optionally, select a certificate from the Certificates dialog. The **Intended Purposes** column must include at least **Client Authentication and Server Authentication**:



Issued to	Issued by	Intended Purposes	Friendl...
	osisoft.com	Client Authentication	None
bodo2pc....	osisoft.com	Client Authentication, Server A...	None
PISQLDAS	PISQLDAS	<All>	None

Note: Enterprise certificates are typically already installed on your computer if it is part of a domain. You should see your organization's name in the **Issued by** column. The certificate named PISQLDAS is the SELF-SIGNED certificate used by default.

3. Click **OK**.

4. This output indicates that the selected certificate has been bound to the SSL port:

   ```
   C:\PIPC\SQLDAS>pisqldasAutoConfig.exe -e
   Found existing binding ......deleted
   Create new SSL binding ......OK
   Updating config file   ......OK
   ```

   In this example, the tool has configured the selected certificate to be used by PI SQL DAS and bound this certificate to IP address/port **0.0.0.0:5461**.

5. Restart the PI SQL Data Access Server service to use the new configuration.

SSL Port

The port used for PI SQL Data Access Server HTTPS communication is **5461**.

The *configuration tool* (page 8) allows communication from any IP address. To further restrict the port you can use operating system tools for this purpose. For details, see the Microsoft Windows documentation on *HttpCfg.exe http://msdn.microsoft.com/en-us/library/aa364478(VS.85).aspx*, *Netsh.exe http://msdn.microsoft.com/en-us/library/bb736546(VS.85).aspx*, or Windows integrated firewall.

You may also have 3rd party Firewall or Virus Scanner tools that can restrict communication through port **5461**.

PI Server Login through PI SQL DAS

For the PI Server login, OSIsoft recommends that you use Integrated Security (SSPI) configuration. The PI SQL Data Access Server will log into the PI Server with the user credentials of its client, that is, the PI JDBC driver connection.

The Windows user of the PI SQL DAS login must have a trust configured on the PI Server or have a PI Server identity mapping configured on the PI Server (available in PI Server 3.4.380 or later). See *Configuring PI Server Security* for details about how to configure trusts.

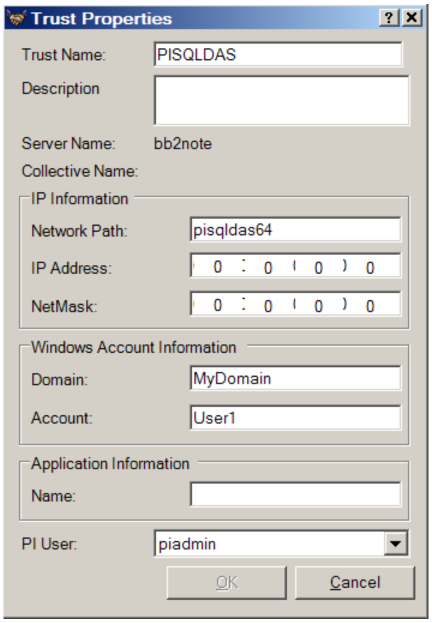
For example, use these properties to configure a trust for Windows user `MyDomain\User1`:

Trust Name: PISQLDAS

Network Path: pisqldas64

Domain: MyDomain

Account: User1



Trust Properties

Trust Name: PISQLDAS

Description:

Server Name: bb2note

Collective Name:

IP Information

Network Path: pisqldas64

IP Address: 0 . 0 . 0 . 0

NetMask: 0 . 0 . 0 . 0

Windows Account Information

Domain: MyDomain

Account: User1

Application Information

Name:

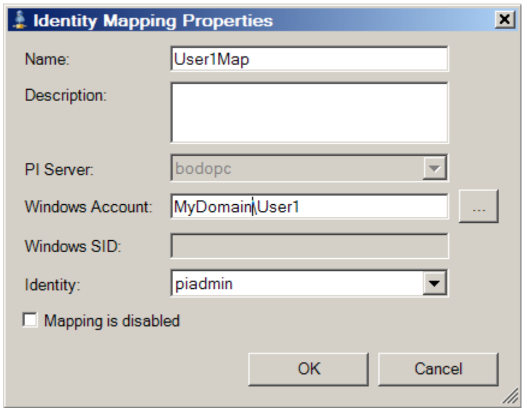
PI User: piadmin

To map a PI Server Identity for Windows user `MyDomain\User1`:

Name: User1Map

Windows Account: MyDomain\User1

Identity: piadmin



Identity Mapping Properties

Name: User1Map

Description:

PI Server: bodopc

Windows Account: MyDomain\User1

Windows SID:

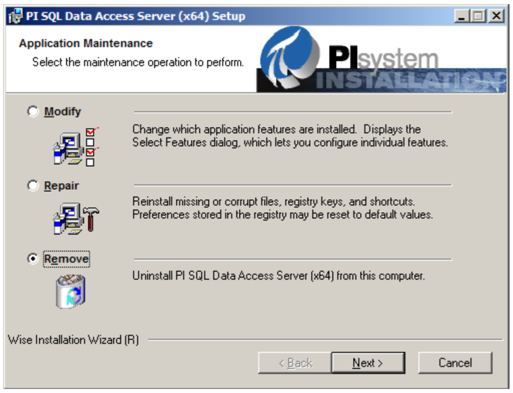
Identity: piadmin

☐ Mapping is disabled

In PI JDBC, the authentication information is configured in the JDBC connection string. For details, see *Using PI JDBC* (page 21). The **getSnap** test application is configured to use the trust mechanism.

Remove PI SQL DAS

To remove the PI SQL Data Access Server, use Windows Control Panel or re-run the setup kit. The **Remove** option automatically stops and deletes the PI SQL DAS service and uninstalls all files:



PI SQL Data Access Server (x64) Setup

Application Maintenance

Select the maintenance operation to perform.

○ Modify

Change which application features are installed. Displays the Select Features dialog, which lets you configure individual features.

○ Repair

Reinstall missing or corrupt files, registry keys, and shortcuts. Preferences stored in the registry may be reset to default values.

● Remove

Uninstall PI SQL Data Access Server (x64) from this computer.

Wise Installation Wizard (R)

The SELF-SIGNED certificate and the SSL port binding are removed when PI SQL DAS is uninstalled.

Install PI JDBC Driver on Windows

1. Run the *PI JDBC driver setup kit* (page 12).
2. To validate the installation results, review the *files installed* (page 13) and the *system environment variables* (page 13).
3. *Verify the installation* (page 14) using sample the Java application **getSnap**.

Run the PI JDBC Driver Setup Kit

To start the PI JDBC Installation Wizard, run PI JDBC_2012_.exe.

Silent Installation

The PI JDBC setup kit extracts several installation modules. The components of the installation process, their order, and the arguments used to launch the components are provided in a configuration file named `setup.ini`. If you modify this file, you can provide different command line arguments for different stages of the setup. This may be useful within a well-controlled environment with options that are known in advance, such as in the case of an embedded installation. The setup kit also contains a file named `silent.ini` that contains modifications to `setup.ini` that are typically needed to run a silent installation. You can augment these arguments by adding any of the options described below.

Individual arguments must contain no spaces unless they are surrounded by quotes.

Argument	Description
/i	Specifies an installation
/qn	Specifies the "quiet mode" and suppresses dialog boxes and prompts
ALLUSERS	Specifies the per-machine or per-user installation context. Use a value of **1** for silent installations.
REBOOT	Restarts the computer. Use a value of **Suppress** for silent installations.

Use this syntax for a silent installation of a single component:

```
msiexec.exe /i PIJDBC.msi REBOOT=Suppress
ALLUSERS=1 /qn
msiexec.exe /i PIJDBC64.msi REBOOT=Suppress
ALLUSERS=1 /qn
```

Note: To run the complete package in silent mode, replace the setup.ini file with silent.ini and run Setup.exe.

Files Installed

After you run the *PI JDBC Driver setup kit* (page 12), you can view the files installed in the `[PIHOME]\JDBC` directory. The complete list of files is provided in the PI JDBC 2012 Release Notes.

Installation Results

1. *Files* (page 13) are installed into the `[PIHOME]\JDBC` directory.
2. The following system environment variables are automatically added/updated:

```
PI_RDSA_LIB=[PIHOME]\JDBC\RDSAWrapper.dll
PI_RDSA_LIB64=[PIHOME]\JDBC\RDSAWrapper64.dll(64-bit only)
CLASSPATH=[PIHOME]\JDBC\PIJDBCDriver.jar;.
```

`[PIHOME]` is replaced by the value of the `PIHOME` directory.

Note: The CLASSPATH environment variable is a standard mechanism in Java to easily launch an application without having to specify the path separately. If the CLASSPATH already existed, the PI JDBC installation appends the name and location of the PI JDBC driver plus a dot (**.**) which denotes the *current directory*.

Verify the PI JDBC Installation

PI JDBC ships with a sample command line application called **getSnap** to validate successful installation and communication with a PI Server. There is also a sample command line application called **getEASnap** to validate successful communication with the PI Asset Framework (AF).

Refer to the information here to run these applications.

getSnap

The **getSnap** test application is located in the Samples directory of the PI JDBC installation. To open it, use a command prompt window:

1. Change to the directory `<PIHOME>\PIPC\JDBC\Samples\getSnap\bin`
2. Enter these three parameters:

- `PI SQL DAS name`
- `PI Server name`
- `PI tag name or tag name wildcard (SQL Syntax)`

For example, if you enter:

```
<PIHOME>\JDBC\Samples\getSnap\bin>java getSnap PISQLDAS01
PIServer01 sin%
```

You should see results in the form of PI data values such as:

```
SINUSOID 79.11583
SINUSOID.Fast 95.47511
SINUSOID_Alert 18.164595
SINUSOID_Fast 95.47511
SINUSOIDU 29.995642
```

Note: On first use you will see the *security login* (page 16) dialog.

getEASnap

The **getEASnap** test application is located in a Samples directory where PI JDBC is installed.

1. To open **getEASnap**, use a command prompt window and change to the directory:

   ```
   <PIHOME>\PIPC\JDBC\Samples\getEASnap\bin
   ```

2. Enter these four parameters:

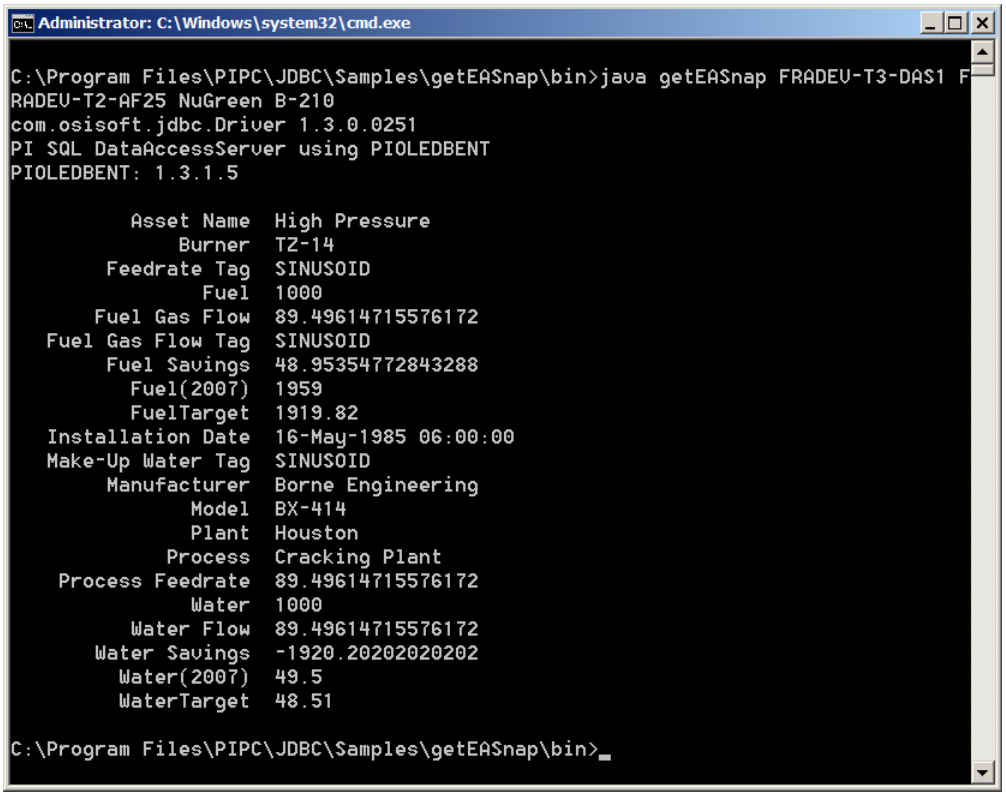
   ```
   PI SQL DAS name
   AF Server name
   AF Database name
   Element name
   ```

   Note: This example assumes that the NuGreen test database, which is shipped with PI OLEDB Enterprise, is loaded.

   For example, if you enter:

   ```
   C:\PIPC\JDBC\Samples\getEASnap\bin>java getEASnap PISQLDAS01
   AFServer01 NuGreen B-210
   ```

   You should see output such as shown here:



```
C:\Program Files\PIPC\JDBC\Samples\getEASnap\bin>java getEASnap FRADEV-T3-DAS1 F
RADEV-T2-AF25 NuGreen B-210
com.osisoft.jdbc.Driver 1.3.0.0251
PI SQL DataAccessServer using PIOLEDBENT
PIOLEDBENT: 1.3.1.5

          Asset Name  High Pressure
              Burner  TZ-14
       Feedrate Tag  SINUSOID
                Fuel  1000
       Fuel Gas Flow  89.49614715576172
   Fuel Gas Flow Tag  SINUSOID
        Fuel Savings  48.95354772843288
          Fuel(2007)  1959
          FuelTarget  1919.82
   Installation Date  16-May-1985 06:00:00
   Make-Up Water Tag  SINUSOID
        Manufacturer  Borne Engineering
               Model  BX-414
               Plant  Houston
             Process  Cracking Plant
    Process Feedrate  89.49614715576172
               Water  1000
          Water Flow  89.49614715576172
       Water Savings  -1920.20202020202
         Water(2007)  49.5
         WaterTarget  48.51

C:\Program Files\PIPC\JDBC\Samples\getEASnap\bin>_
```

Note: On first use you will see the *security login* (page 16) dialog.

Security

Two logins are required to connect to the PI Server through PI SQL DAS:

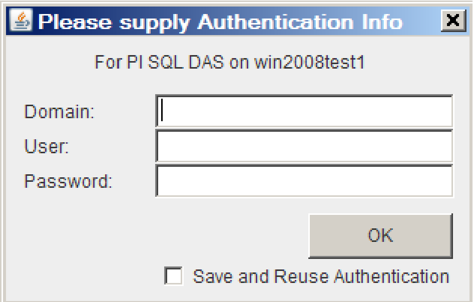
- A *login* (page 16) to the PI SQL Data Access Server
- A *login* (page 16) to the PI Server

Connect with PI SQL Data Access Server

The JDBC login mechanism is used to connect with the PI SQL Data Access Server.

After you install PI JDBC and attempt the first connection with PI SQL Data Access Server, use the login dialog to:

1. Provide valid Windows user login information for the machine that runs PI SQL DAS:



Please supply Authentication Info

For PI SQL DAS on win2008test1

Domain:

User:

Password:

☐ Save and Reuse Authentication

2. To store the login credentials, select **Save and Reuse Authentication**. Login information is stored in an encrypted file in your home directory (%USERPROFILE%). The filename consists of your PI SQL DAS server name suffixed by **.dca**. For example, **win2008test1.dca**. Subsequent logins of the same user automatically refer to that file until it is manually deleted.
3. Click **OK**.

PI Server Login

For the PI Server login, OSIsoft recommends that you use Integrated Security (SSPI) configuration. For complete details, see *Configuring PI Server Security*, available at the *OSIsoft Technical Support Web site* (*http://techsupport.osisoft.com*).

In PI JDBC, the authentication information is configured in the JDBC connection string. For details, see *Using PI JDBC* (page 21). The getSnap test application is configured to use the trust mechanism.

Remove PI JDBC

To remove PI JDBC on Windows, use Windows Control Panel or re-run the setup kit.

Install PI JDBC Driver on Linux

To install PI JDBC driver on Linux operating systems:

1. Select either the *standalone* (page 4) or *middleware* (page 4) deployment option.
2. Verify whether the computer you will use meets the *System Requirements* (page 5).
3. Remove any previous version of PI JDBC.
4. Extract and run the *PI JDBC driver setup kit* (page 17).
5. Add PI JDBC driver location to *Java* `CLASSPATH [optional]` (page 20).

Note: The process to install PI JDBC Driver on Linux are very similar to the *procedures* (page 12) used for Windows operating systems.

Set up PI JDBC Driver on Linux

Extract contents of PI JDBC_2012-x86-x64_.tar.gz.

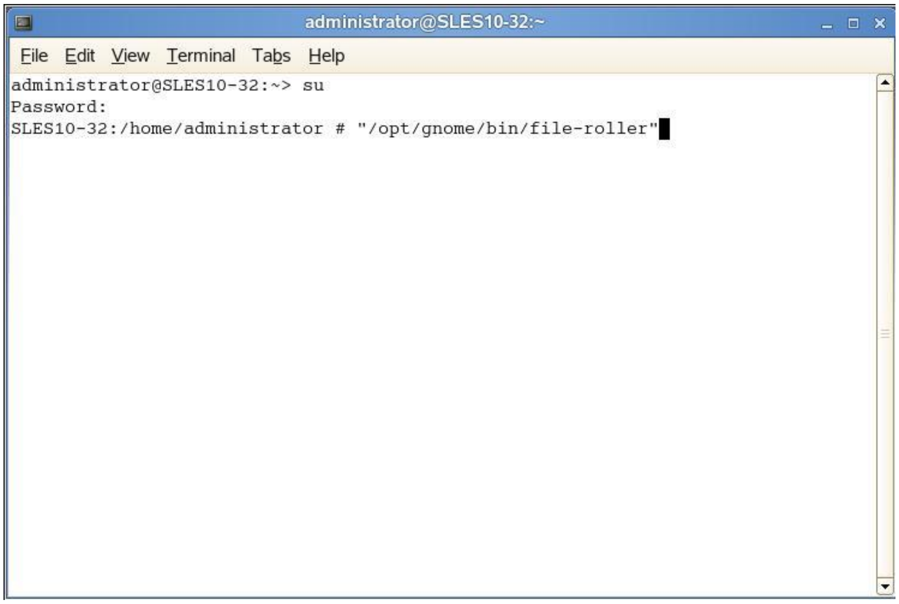
Note: The same setup file is used for 32-bit and 64-bit Linux versions.

OSIsoft recommends that the installation files be installed into the `/opt` directory, but you can choose any other directory as needed.

The different Linux distributions have different tools to execute this task. Below is an example performed on **SLES10**.

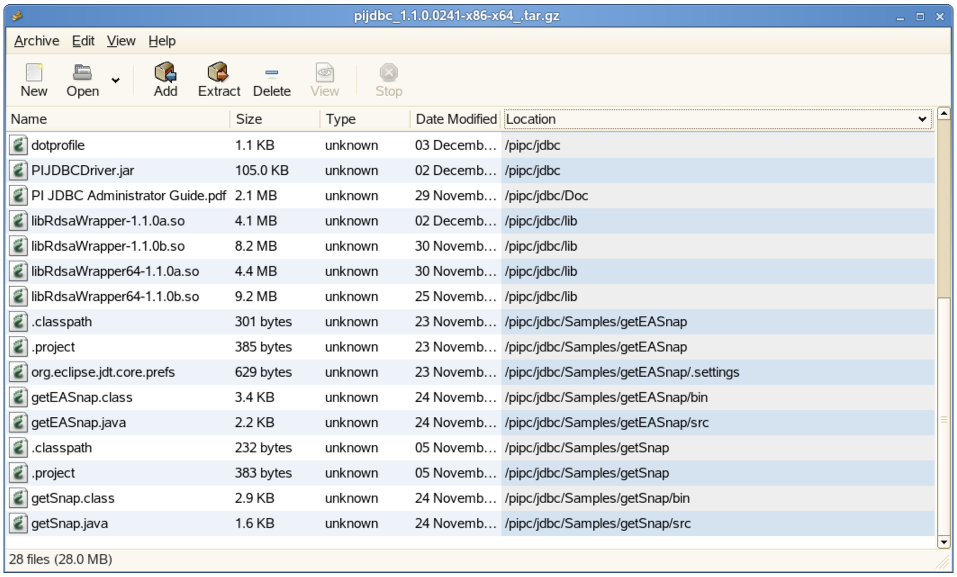
1. To launch File Roller with root privileges, enter in a command prompt:

   ```
   administrator@SLES10-32:~> su
   Password:
   SLES10-32:/home/administrator # "/opt/gnome/bin/file-roller"
   ```



```
administrator@SLES10-32:~> su
Password:
SLES10-32:/home/administrator # "/opt/gnome/bin/file-roller"
```

2. Click **Extract** to extract the files:



Name	Size	Type	Date Modified	Location
dotprofile	1.1 KB	unknown	03 Decemb…	/pipc/jdbc
PIJDBCDriver.jar	105.0 KB	unknown	02 Decemb…	/pipc/jdbc
PI JDBC Administrator Guide.pdf	2.1 MB	unknown	29 Novemb…	/pipc/jdbc/Doc
libRdsaWrapper-1.1.0a.so	4.1 MB	unknown	02 Decemb…	/pipc/jdbc/lib
libRdsaWrapper-1.1.0b.so	8.2 MB	unknown	30 Novemb…	/pipc/jdbc/lib
libRdsaWrapper64-1.1.0a.so	4.4 MB	unknown	30 Novemb…	/pipc/jdbc/lib
libRdsaWrapper64-1.1.0b.so	9.2 MB	unknown	25 Novemb…	/pipc/jdbc/lib
.classpath	301 bytes	unknown	23 Novemb…	/pipc/jdbc/Samples/getEASnap
.project	385 bytes	unknown	23 Novemb…	/pipc/jdbc/Samples/getEASnap
org.eclipse.jdt.core.prefs	629 bytes	unknown	23 Novemb…	/pipc/jdbc/Samples/getEASnap/.settings
getEASnap.class	3.4 KB	unknown	24 Novemb…	/pipc/jdbc/Samples/getEASnap/bin
getEASnap.java	2.2 KB	unknown	24 Novemb…	/pipc/jdbc/Samples/getEASnap/src
.classpath	232 bytes	unknown	05 Novemb…	/pipc/jdbc/Samples/getSnap
.project	383 bytes	unknown	05 Novemb…	/pipc/jdbc/Samples/getSnap
getSnap.class	2.9 KB	unknown	24 Novemb…	/pipc/jdbc/Samples/getSnap/bin
getSnap.java	1.6 KB	unknown	24 Novemb…	/pipc/jdbc/Samples/getSnap/src

28 files (28.0 MB)

Note: If you cannot gain root privileges, OSIsoft recommends that you instead extract the files to your home directory.

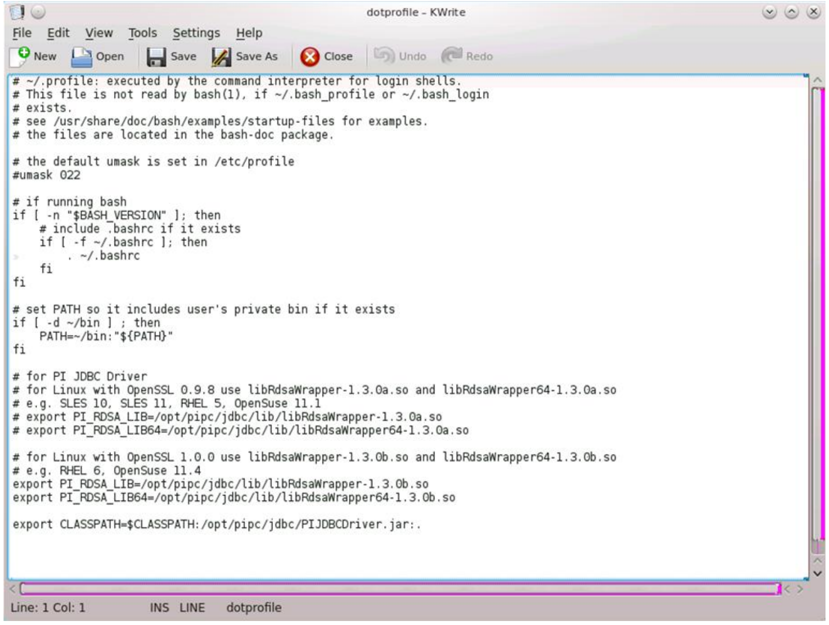
Files Installed

After you run the *PI JDBC Driver setup kit* (page 12), you can view the files installed in the `[PIHOME]\JDBC` directory. The complete list of files is provided in the PI JDBC 2012 Release Notes.

System Variable PI_RDSA_LIB

PI JDBC requires properly defined system variables. There are various mechanisms you can use to complete this task.

OSIsoft recommends that variables be defined in the user profile or system profile. In this example, the system variables are defined in the user profile:



```
# ~/.profile: executed by the command interpreter for login shells.
# This file is not read by bash(1), if ~/.bash_profile or ~/.bash_login
# exists.
# see /usr/share/doc/bash/examples/startup-files for examples.
# the files are located in the bash-doc package.

# the default umask is set in /etc/profile
#umask 022

# if running bash
if [ -n "$BASH_VERSION" ]; then
    # include .bashrc if it exists
    if [ -f ~/.bashrc ]; then
        . ~/.bashrc
    fi
fi

# set PATH so it includes user's private bin if it exists
if [ -d ~/bin ] ; then
    PATH=~/bin:"${PATH}"
fi

# for PI JDBC Driver
# for Linux with OpenSSL 0.9.8 use libRdsaWrapper-1.3.0a.so and libRdsaWrapper64-1.3.0a.so
# e.g. SLES 10, SLES 11, RHEL 5, OpenSuse 11.1
# export PI_RDSA_LIB=/opt/pipc/jdbc/lib/libRdsaWrapper-1.3.0a.so
# export PI_RDSA_LIB64=/opt/pipc/jdbc/lib/libRdsaWrapper64-1.3.0a.so

# for Linux with OpenSSL 1.0.0 use libRdsaWrapper-1.3.0b.so and libRdsaWrapper64-1.3.0b.so
# e.g. RHEL 6, OpenSuse 11.4
export PI_RDSA_LIB=/opt/pipc/jdbc/lib/libRdsaWrapper-1.3.0b.so
export PI_RDSA_LIB64=/opt/pipc/jdbc/lib/libRdsaWrapper64-1.3.0b.so

export CLASSPATH=$CLASSPATH:/opt/pipc/jdbc/PIJDBCDriver.jar:.
```

Line: 1 Col: 1   INS LINE   dotprofile

- For OpenSSL 0.9.8 use:

  ```
  export PI_RDSA_LIB=/opt/pipc/jdbc/lib/libRdsaWrapper-1.3.0a.so
  export PI_RDSA_LIB64=/opt/pipc/jdbc/lib/libRdsaWrapper64-1.3.0a.so
  ```

- For OpenSSL 1.0.0 use

  ```
  export PI_RDSA_LIB=/opt/pipc/jdbc/lib/libRdsaWrapper-1.3.0b.so
  export PI_RDSA_LIB64=/opt/pipc/jdbc/lib/libRdsaWrapper64-1.3.0b.so
  ```

Note: On 32-bit Linux, the 64-bit PI JDBC driver cannot be called, therefore you do not need to define **PI_RDSA_LIB64**.

Validating Installation

PI JDBC ships with a sample application called **getSnap** to validate successful installation. To use **getSnap**:

1. *Add the driver to the CLASSPATH Variable* (page 20).
2. Enter these three parameters:

   ```
   PI SQL DAS name
   PI Server name
   PI tag name or tag name wildcard (SQL Syntax)
   ```

   For example, if you enter:

   ```
   $java getSnap PISQLDAS01 PIServer01 sin%
   ```

   You should see results in the form of PI data values such as:

   ```
   SINUSOID 79.11583
   SINUSOID.Fast 95.47511
   SINUSOID_Alert 18.164595
   SINUSOID_Fast 95.47511
   SINUSOIDU 29.995642
   ```

Note: On first use you will see the *security login* (page 16) dialog.

Add the Driver to the CLASSPATH Variable

In order to easily launch a PI JDBC command line application we recommend you add the PI JDBC driver to the CLASSPATH environment variable.

In Linux this is done with an export command as shown for PI_RDSA_LIB:

```
export CLASSPATH=$CLASSPATH:/opt/pipc/jdbc/PIJDBCDriver.jar:.
```

getEASnap

To use the **getEASnap** test application to verify your installation, see *getEASnap* (page 14).

Remove PI JDBC (Linux)

1. Remove your `PI_RDSA_LIB`, `PI_RDSA_LIB64` and `CLASSPATH` modification.
2. Delete all files in the `/pipc/jdbc` directory.

Chapter 3

Using PI JDBC

Configuration

Driver Class

Before a JDBC driver can make a connection it needs to be registered with the Driver Manager. Applications typically either prompt you for the location of the driver and then auto discover the JDBC driver name, or directly prompt you to configure the driver name (class name).

The driver name for PI JDBC is:

```
com.osisoft.jdbc.Driver
```

URL Format

To connect to a database, JDBC uses database URLs. The URL for PI JDBC needs to provide the name of the PI SQL DAS machine and the PI OLEDB Enterprise/PI OLEDB connection string.

- To use PI OLEDB Enterprise table set and connect to the PI Asset Framework (AF):

  ```
  jdbc:pi//pisqldas_server/pioledbent_connectionstring
  ```

  - For example:

  ```
  jdbc:pi//mySQLDAS/Data Source=myAFServer; Integrated
  Security=SSPI;
  ```

- To use PI OLEDB Provider and connect to the PI Archive:

  ```
  jdbc:pisql://pisqldas_server/pioledb_connectionstring
  ```

- For example:

  ```
  jdbc:pisql://mySQLDAS/Data Source=myPIServer; Integrated
  Security=SSPI;
  ```

Time Zone

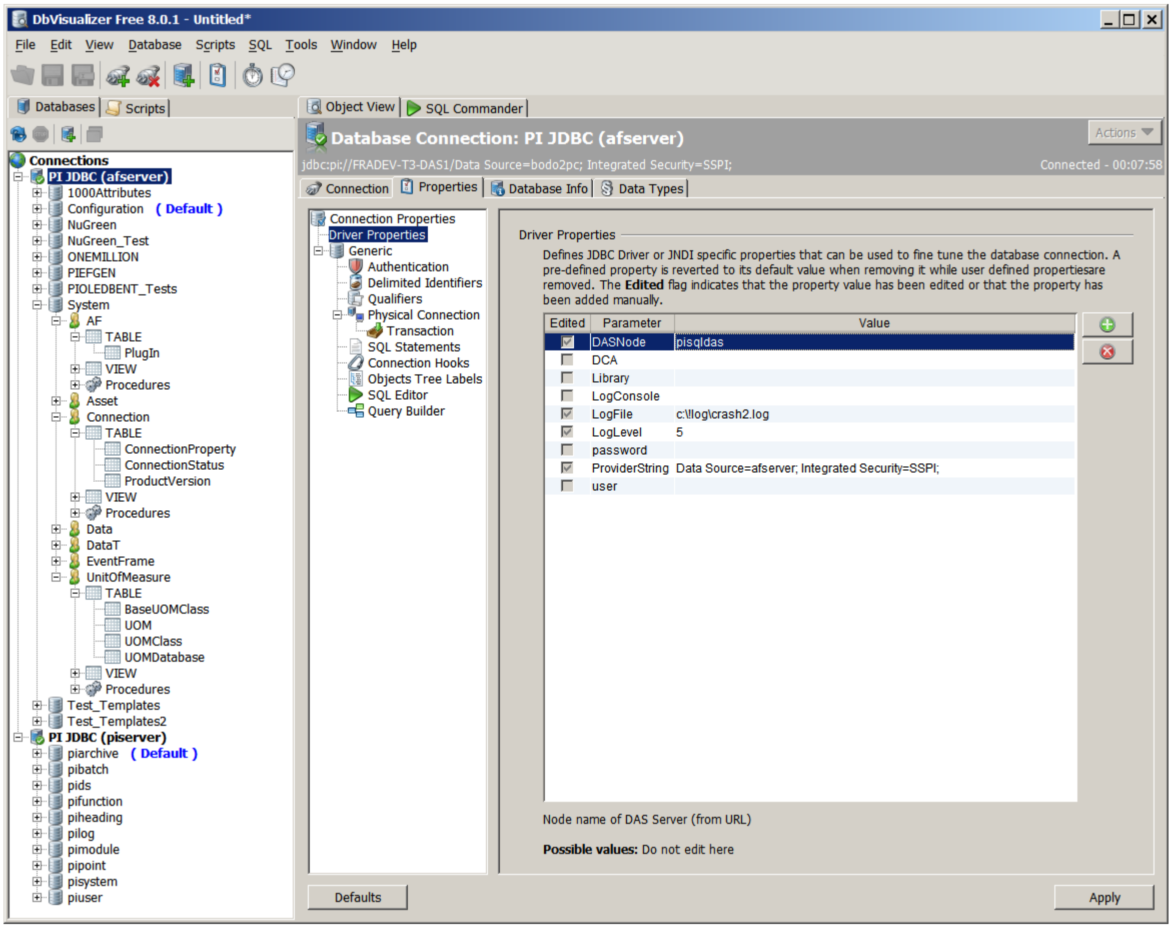
In this version Time Zone/DST settings have to be the same for the PI SQL DAS computer and the PI JDBC client computer.

PI JDBC automatically uses `"Time Zone=Local;"`.

Driver Properties

Driver properties are used to specify details for making a connection to the underlying database. In general driver properties can be required or optional. PI JDBC has only optional properties. Even username and password are optional because there is an alternative way to specify authentication information (see DCA property in the table below).

Optional driver properties can be specified in addition to the URL string and typically control additional driver features such as logging capabilities and connection timeouts. These driver properties can only be specified if exposed by the JDBC client or from a development environment. For example, here driver properties are exposed by DBVisualizer:



Database Connection: PI JDBC (afserver)

jdbc:pi://FRADEV-T3-DAS1/Data Source=bodo2pc; Integrated Security=SSPI;

Connected - 00:07:58

Driver Properties

Defines JDBC Driver or JNDI specific properties that can be used to fine tune the database connection. A pre-defined property is reverted to its default value when removing it while user defined propertiesare removed. The **Edited** flag indicates that the property value has been edited or that the property has been added manually.

Edited	Parameter	Value
☑	DASNode	pisqldas
☐	DCA	
☐	Library	
☐	LogConsole	
☑	LogFile	c:\!log\crash2.log
☑	LogLevel	5
☐	password	
☑	ProviderString	Data Source=afserver; Integrated Security=SSPI;
☐	user	

Node name of DAS Server (from URL)

Possible values: Do not edit here

Driver Property	Description	Examples
DASNode	Name of PI SQL DAS node. Is automatically populated from URL. **DO NOT EDIT**	
DCA	Options to ignore or reuse persisted authentication (`filename.dca`) for PI SQL DAS The DCA file is specific to a certain user and a certain PI SQL DAS Values: INIT – clears persisted authentication information REUSE – reuses persisted authentication file name – use specific authentication file SAVE – persists a DCA file from values specified in user and password properties **Note:** The value **SAVE** can be used to create a DCA file in an environment without graphical user interface	**DCA=C:\PIPC\JDBC\myAuthentication.dca** **DCA=INIT** **DCA=REUSE (default)**
LogConsole	Print log messages to Console.	**LogConsole=True**
LogFile	Full log file path/name.	**LogFile = C:\Temp\Log\PIJDBC.log**
LogLevel	Specifies granularity of messages to be logged. Values: **0-5**	**LogLevel=3**
password	Password for PI SQL DAS login. Automatically filled from Logon screen or persisted authentication file.	
ProviderString	PI OLEDB or PI OLEDB Enterprise provider string. Automatically filled from URL. **DO NOT EDIT**	
user	Username for PI SQL DAS login. Automatically filled from Login dialog or persisted authentication file.	

Functionality

SQL

PI JDBC delegates all SQL commands via PI SQL DAS to the underlying query engine.

When connected to Asset Framework (AF) Server, the built-in query engine of PI OLEDB Enterprise is used. Please refer to the *PI OLEDB Enterprise User Guide* for supported SQL syntax. When connected to a PI Server, the built-in query engine of PI OLEDB is used. Refer to the *PI OLEDB Provider User Manual* for supported SQL syntax.

PI OLEDB documentation is available at the *OSIsoft Technical Support Web site* (*http://techsupport.osisoft.com*).

Advanced JDBC

Multithreading

PI JDBC supports query execution from multiple threads. We recommend reusing connection objects as much as possible because connection open and close operations are expensive.

For example, connection objects can be passed as parameter to a new thread and when that thread is finished, it gives that connection to a connection pool.

Batch Processing

It is possible to assign a series of SQL statements to a JDBC statement to be submitted for execution by PI SQL DAS. This mechanism can be used to:

- aggregate multiple executions of a parameterized query, such as INSERT statements
- execute multiple independent queries at once (these queries cannot have parameters)

JDBC Driver Extensions

PI JDBC supports JDBC extensions, also known as JDBC 2.0 Optional Package, or `javax.sql`, that allow configuring connections in a directory service through JNDI. It also allows using JNDI connection pooling.

Note: The connection pooling is a dummy implementation to support environments that demand that interface. However, there is no performance gain when using this interface. OSIsoft therefore recommends reuse of connections in the application itself. PI Server and AF Server connections are pooled within PI SQL DAS.

Message Logging

PI SQL DAS

PI SQL DAS puts errors and other messages into the local PI SDK log. This local log is where PI SQL DAS runs, not where PI JDBC runs.

The messages are retrievable via the PI JDBC driver by querying the new `pilog..pisdklog` table, introduced in PI OLEDB 3.3.0.1.

Note: If the PI JDBC driver has connection problems and you cannot execute a `pilog..pisdklog` query, then use PI OLEDB to get to this same information.

PI JDBC

The PI JDBC driver offers two logging mechanisms:

- Remote configuration of PI OLEDB logging to get details about query execution. PI OLEDB logging is configurable via the connection string part of the URL.

  Note: For security reasons the **Log File** keyword is disabled for remote use. Use the **Server Log** keyword instead.

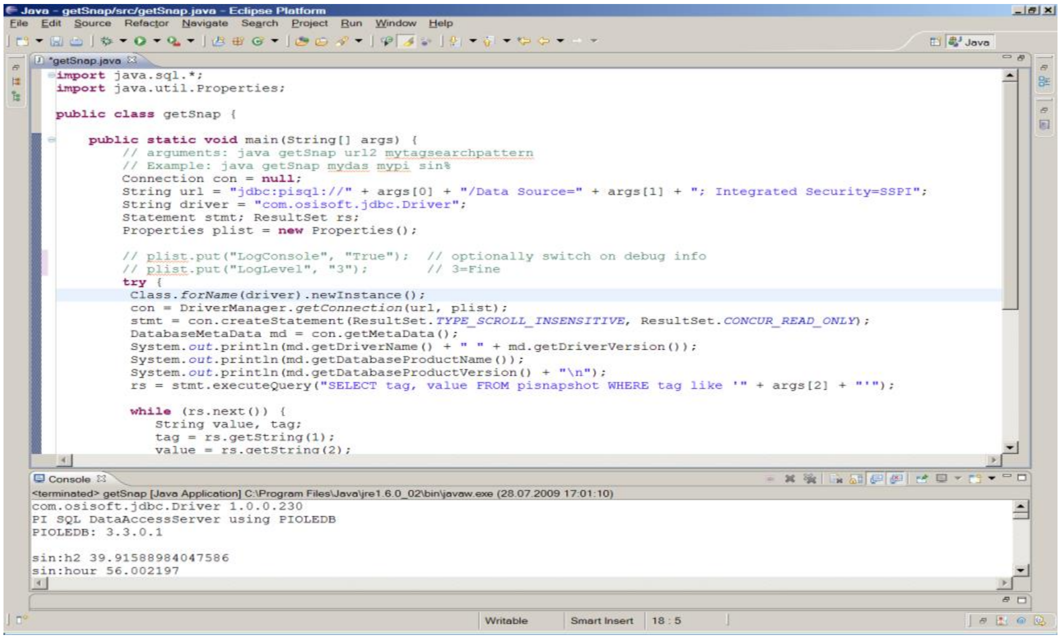
- Local PI JDBC log file.

Use Cases

Source Code Example

The source code for the **getSnap** test application is located in the Samples directory of the Windows version.

For example, you can execute getSnap in the Eclipse IDE development platform to step through the code as shown below.



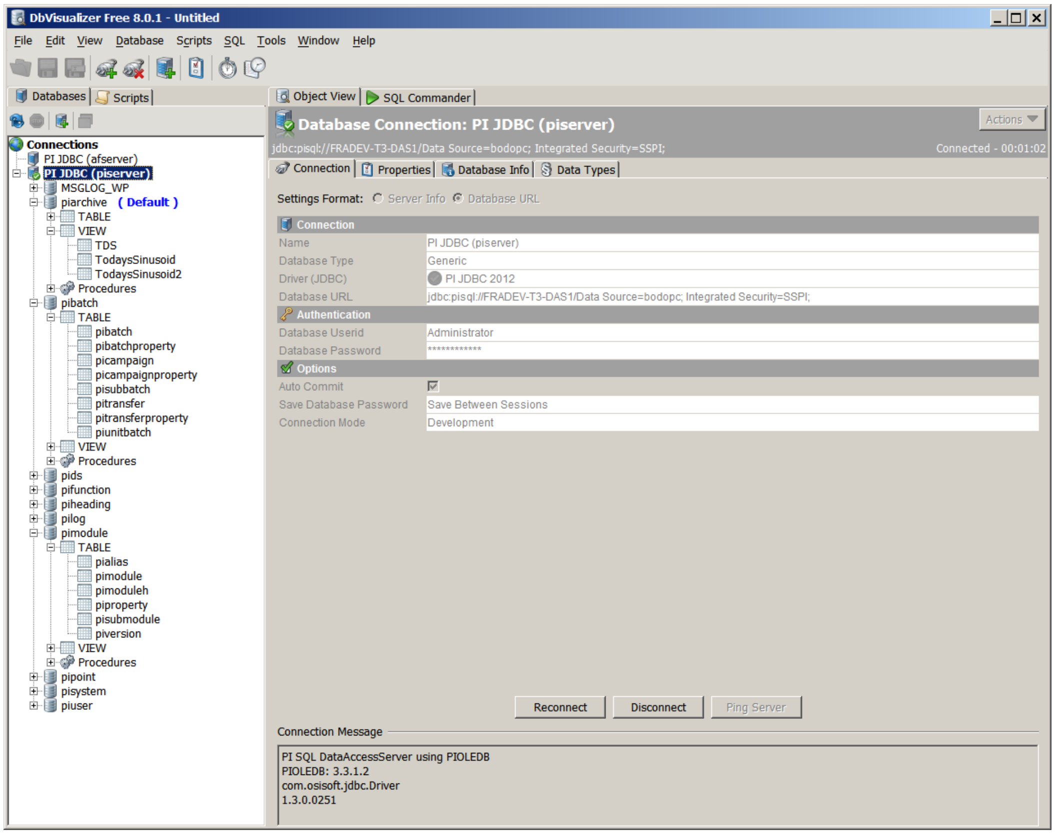
Java - getSnap/src/getSnap.java - Eclipse Platform
File Edit Source Refactor Navigate Search Project Run Window Help
Java
*getSnap.java
Console
Writable
Smart Insert
18 : 5

```
import java.sql.*;
import java.util.Properties;

public class getSnap {

    public static void main(String[] args) {
        // arguments: java getSnap url2 mytagsearchpattern
        // Example: java getSnap mydas mypi sin%
        Connection con = null;
        String url = "jdbc:pisql://" + args[0] + "/Data Source=" + args[1] + "; Integrated Security=SSPI";
        String driver = "com.osisoft.jdbc.Driver";
        Statement stmt; ResultSet rs;
        Properties plist = new Properties();

        // plist.put("LogConsole", "True");  // optionally switch on debug info
        // plist.put("LogLevel", "3");       // 3=Fine
        try {
         Class.forName(driver).newInstance();
         con = DriverManager.getConnection(url, plist);
         stmt = con.createStatement(ResultSet.TYPE_SCROLL_INSENSITIVE, ResultSet.CONCUR_READ_ONLY);
         DatabaseMetaData md = con.getMetaData();
         System.out.println(md.getDriverName() + " " + md.getDriverVersion());
         System.out.println(md.getDatabaseProductName());
         System.out.println(md.getDatabaseProductVersion() + "\n");
         rs = stmt.executeQuery("SELECT tag, value FROM pisnapshot WHERE tag like '" + args[2] + "'");

         while (rs.next()) {
            String value, tag;
            tag = rs.getString(1);
            value = rs.getString(2);
```

```
<terminated> getSnap [Java Application] C:\Program Files\Java\jre1.6.0_02\bin\javaw.exe (28.07.2009 17:01:10)
com.osisoft.jdbc.Driver 1.0.0.230
PI SQL DataAccessServer using PIOLEDB
PIOLEDB: 3.3.0.1

sin:h2 39.91588984047586
sin:hour 56.002197
```

DBVisualizer

If you want to explore the PI JDBC functionality without using a programming environment, you might consider using **DBVisualizer** *http://www.dbvis.com* (*http://www.dbvis.com/products/dbvis/*):



Database Connection: PI JDBC (piserver)

jdbc:pisql://FRADEV-T3-DAS1/Data Source=bodopc; Integrated Security=SSPI;

Connected - 00:01:02

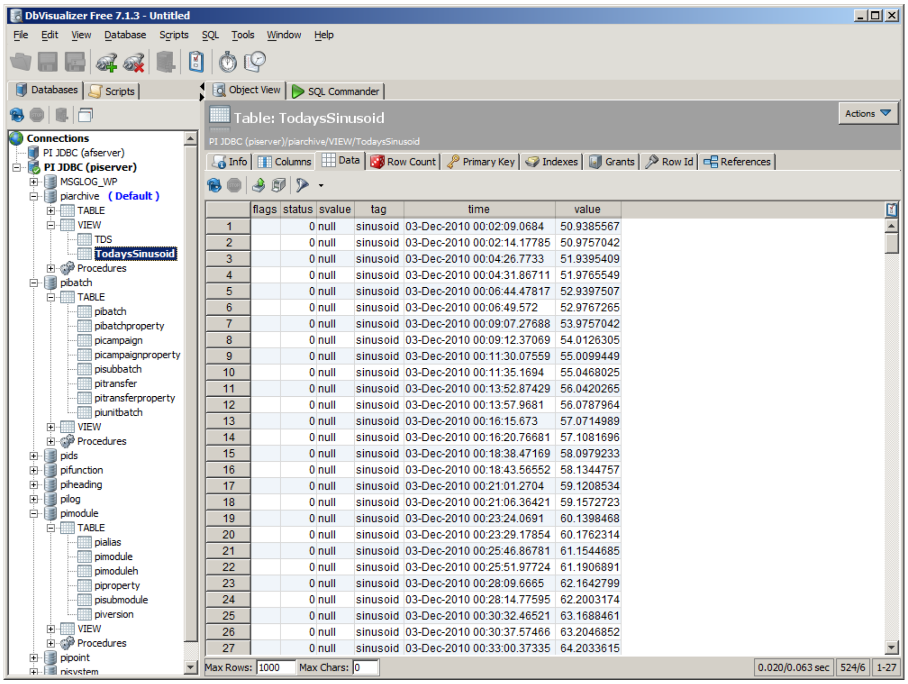
Settings Format: ○ Server Info ● Database URL

Connection	
Name	PI JDBC (piserver)
Database Type	Generic
Driver (JDBC)	PI JDBC 2012
Database URL	jdbc:pisql://FRADEV-T3-DAS1/Data Source=bodopc; Integrated Security=SSPI;
Authentication	
Database Userid	Administrator
Database Password	************
Options	
Auto Commit	☑
Save Database Password	Save Between Sessions
Connection Mode	Development

Connection Message

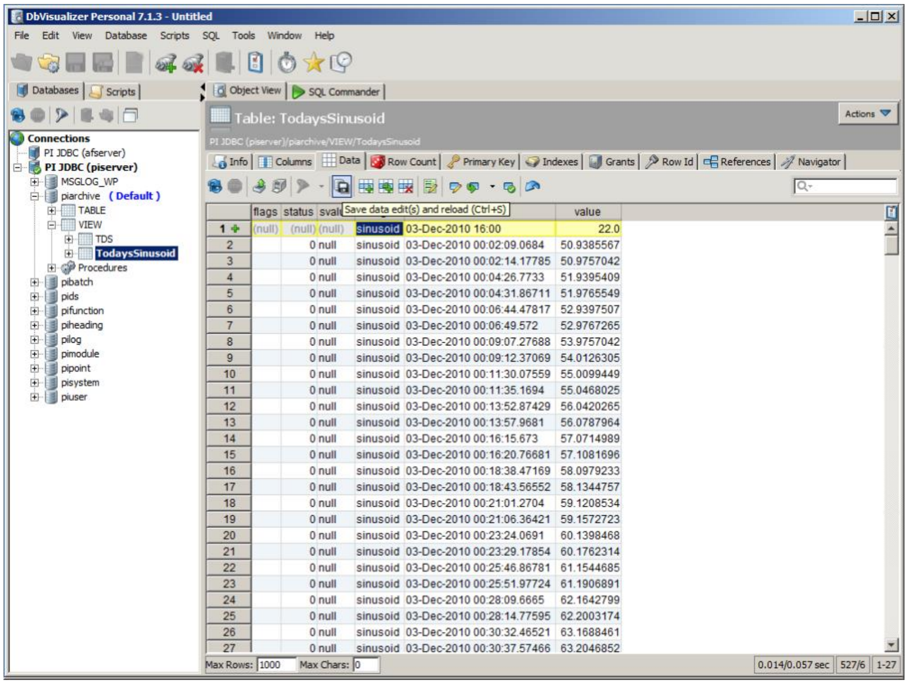
```
PI SQL DataAccessServer using PIOLEDB
PIOLEDB: 3.3.1.2
com.osisoft.jdbc.Driver
1.3.0.0251
```

This figure illustrates how you can open a PI SQL View using DBVisualizer:



	flags	status	svalue	tag	time	value
1		0	null	sinusoid	03-Dec-2010 00:02:09.0684	50.9385567
2		0	null	sinusoid	03-Dec-2010 00:02:14.17785	50.9757042
3		0	null	sinusoid	03-Dec-2010 00:04:26.7733	51.9395409
4		0	null	sinusoid	03-Dec-2010 00:04:31.86711	51.9765549
5		0	null	sinusoid	03-Dec-2010 00:06:44.47817	52.9397507
6		0	null	sinusoid	03-Dec-2010 00:06:49.572	52.9767265
7		0	null	sinusoid	03-Dec-2010 00:09:07.27688	53.9757042
8		0	null	sinusoid	03-Dec-2010 00:09:12.37069	54.0126305
9		0	null	sinusoid	03-Dec-2010 00:11:30.07559	55.0099449
10		0	null	sinusoid	03-Dec-2010 00:11:35.1694	55.0468025
11		0	null	sinusoid	03-Dec-2010 00:13:52.87429	56.0420265
12		0	null	sinusoid	03-Dec-2010 00:13:57.9681	56.0787964
13		0	null	sinusoid	03-Dec-2010 00:16:15.673	57.0714989
14		0	null	sinusoid	03-Dec-2010 00:16:20.76681	57.1081696
15		0	null	sinusoid	03-Dec-2010 00:18:38.47169	58.0979233
16		0	null	sinusoid	03-Dec-2010 00:18:43.56552	58.1344757
17		0	null	sinusoid	03-Dec-2010 00:21:01.2704	59.1208534
18		0	null	sinusoid	03-Dec-2010 00:21:06.36421	59.1572723
19		0	null	sinusoid	03-Dec-2010 00:23:24.0691	60.1398468
20		0	null	sinusoid	03-Dec-2010 00:23:29.17854	60.1762314
21		0	null	sinusoid	03-Dec-2010 00:25:46.86781	61.1544685
22		0	null	sinusoid	03-Dec-2010 00:25:51.97724	61.1906891
23		0	null	sinusoid	03-Dec-2010 00:28:09.6665	62.1642799
24		0	null	sinusoid	03-Dec-2010 00:28:14.77595	62.2003174
25		0	null	sinusoid	03-Dec-2010 00:30:32.46521	63.1688461
26		0	null	sinusoid	03-Dec-2010 00:30:37.57466	63.2046852
27		0	null	sinusoid	03-Dec-2010 00:33:00.37335	64.2033615

This figure illustrates how you can insert data into a PI SQL View using DBVisualizer:


DbVisualizer Personal 7.1.3 - Untitled
File Edit View Database Scripts SQL Tools Window Help
Databases | Scripts
Object View | SQL Commander
Connections
PI JDBC (afserver)
PI JDBC (piserver)
MSGLOG_WP
piarchive (Default)
TABLE
VIEW
TDS
TodaysSinusoid
Procedures
pibatch
pids
pifunction
piheading
pilog
pimodule
pipoint
pisystem
piuser
Table: TodaysSinusoid
PI JDBC (piserver)/piarchive/VIEW/TodaysSinusoid
Actions
Info | Columns | Data | Row Count | Primary Key | Indexes | Grants | Row Id | References | Navigator
Save data edit(s) and reload (Ctrl+S)
Max Rows: 1000 Max Chars: 0
0.014/0.057 sec | 527/6 | 1-27

	flags	status	svalue	tag	time	value
1 ✚	(null)	(null)	(null)	sinusoid	03-Dec-2010 16:00	22.0
2		0	null	sinusoid	03-Dec-2010 00:02:09.0684	50.9385567
3		0	null	sinusoid	03-Dec-2010 00:02:14.17785	50.9757042
4		0	null	sinusoid	03-Dec-2010 00:04:26.7733	51.9395409
5		0	null	sinusoid	03-Dec-2010 00:04:31.86711	51.9765549
6		0	null	sinusoid	03-Dec-2010 00:06:44.47817	52.9397507
7		0	null	sinusoid	03-Dec-2010 00:06:49.572	52.9767265
8		0	null	sinusoid	03-Dec-2010 00:09:07.27688	53.9757042
9		0	null	sinusoid	03-Dec-2010 00:09:12.37069	54.0126305
10		0	null	sinusoid	03-Dec-2010 00:11:30.07559	55.0099449
11		0	null	sinusoid	03-Dec-2010 00:11:35.1694	55.0468025
12		0	null	sinusoid	03-Dec-2010 00:13:52.87429	56.0420265
13		0	null	sinusoid	03-Dec-2010 00:13:57.9681	56.0787964
14		0	null	sinusoid	03-Dec-2010 00:16:15.673	57.0714989
15		0	null	sinusoid	03-Dec-2010 00:16:20.76681	57.1081696
16		0	null	sinusoid	03-Dec-2010 00:18:38.47169	58.0979233
17		0	null	sinusoid	03-Dec-2010 00:18:43.56552	58.1344757
18		0	null	sinusoid	03-Dec-2010 00:21:01.2704	59.1208534
19		0	null	sinusoid	03-Dec-2010 00:21:06.36421	59.1572723
20		0	null	sinusoid	03-Dec-2010 00:23:24.0691	60.1398468
21		0	null	sinusoid	03-Dec-2010 00:23:29.17854	60.1762314
22		0	null	sinusoid	03-Dec-2010 00:25:46.86781	61.1544685
23		0	null	sinusoid	03-Dec-2010 00:25:51.97724	61.1906891
24		0	null	sinusoid	03-Dec-2010 00:28:09.6665	62.1642799
25		0	null	sinusoid	03-Dec-2010 00:28:14.77595	62.2003174
26		0	null	sinusoid	03-Dec-2010 00:30:32.46521	63.1688461
27		0	null	sinusoid	03-Dec-2010 00:30:37.57466	63.2046852

PI JDBC Driver Limitations

Time Zone

In this version Time Zone/DST settings have to be the same for the PI SQL DAS computer and the PI JDBC client computer.

Moreover, the PI OLEDB-specific connection property **Time Zone** cannot be used. If specified, the setting has no effect.

Data Types

The following data types are unsupported:

- .NET (AF) Object
- COM Object, etc.
- VARIANT containing GUID or
- VARIANT containing Timestamp

Multiple Connections

Multiple connections from one PI JDBC application to different PI SQL DAS (installed on multiple nodes) are supported.

Transactions

Manual transactions are not supported. Queries always have to be executed in auto-commit mode.

PI Initialization Properties

The following Initialization Properties are preconfigured and cannot be changed (their use in the connection string part of the URL has no effect):

Connection String Keyword	PI OLEDB Enterprise/PI OLEDB Description	Value
Always Return Rowset	If set, all SQL commands return rowsets containing number of rows affected by the execution.	**Always Return Rowset = False**;
Time Zone	Time zone used for parsing of timestamp literals and for rendering of timestamp column values.	**Time Zone = Local**;
Log File	Full log file path/name.	**Log File =**;

The following Initialization Properties have different default values:

Connection String Keyword	PI OLEDB Description	Value
Session ID	Can be a positive integer value or **-1**. Only applies to consumer processes that work with multiple session objects. Allows for grouping of server connections. All sessions with the same positive Session ID belong to the same group. Server calls within the same group are serialized.	**Session ID = -1**; sessions in the same process have own server connections and own cache

Chapter 4

Troubleshooting

To identify solutions when PI JDBC driver does not connect or does not get data, OSIsoft recommends that you troubleshoot from the ground up, and test components in the order they are used:

- PI SDK, check connectivity using **About PI-SDK**
- PI OLEDB Enterprise, verify functionality using **PI SQL Commander**
- PI OLEDB, verify functionality using **PI OLEDB Tester**
- PI SQL Data Access Server, run PI SQL DAS interactively to verify functionality or use a PI OLEDB query in the `pilog..pisdklog` table to check for PI SQL DAS error messages
- PI JDBC, start with installation on same computer as PI SQL DAS

Refer to *Connection Messages and Errors* (page 31) for more troubleshooting information.

Run PI SQL DAS Interactively

To run PI SQL DAS in interactive mode, enter into a command prompt:

```
C:\PIPC\SQLDAS>pisqldas.exe -x
BaseAddress = http://localhost:5460/DataAccessServer
BaseAddress = net.tcp://localhost:5462/DataAccessServer
 EndPoints provided :
    net.tcp://localhost:5462/DataAccessServer/Query
    https://localhost:5461/DataAccessServer/Query

 http://localhost:5460/DataAccessServer
```

Connection Messages and Errors

This topic contains common connection messages, including messages that indicate connection problems and the corresponding error messages.

These messages are visible as output of the getSnap sample application. Other JDBC applications might display messages in a message box or log file, depending on how those applications handle exceptions.

For example, *DBVisualizer* (page 26) displays messages in the **Connection Message** text box of its **Database Connection** dialog.

Successful Connection

```
com.osisoft.jdbc.Driver 1.3.0.0251 (Release)
PI SQL DataAccessServer using PIOLEDB
PIOLEDB: 3.3.0.1

or

com.osisoft.jdbc.Driver 1.3.0.0251 (Release)
PI SQL DataAccessServer using PIOLEDBENT
PIOLEDBENT: 1.3.1.5
```

Incorrect Username/Password for Connection to PI SQL DAS

```
An error occurred while establishing the connection:
   Type: java.sql.SQLException   Error Code: 0
Message:
 [PI SQL DAS gSOAP Channel] HTTP Error
Details: HTTP/1.1 403 Forbidden
```

No SSPI Configuration or Incorrect Username/Password

```
An error occurred while establishing the connection:
   Type: java.sql.SQLException   Error Code: 0
Message:
   [PISQLDAS] [PIOLEDB]
[PI SDK] Unable to open a session on a server.  The user name and
password may be incorrect.  mypiserver
```

Incorrect Data Source (Server) Name

```
An error occurred while establishing the connection:
   Type: java.sql.SQLException   Error Code: 0
Message:
[PISQLDAS] [PIOLEDB]
[PI SDK] The requested server was not found in the known servers
table. Unable to resolve name to IP address. mypi3

or,

An error occurred while establishing the connection:
   Type: java.sql.SQLException   Error Code: 0
Message:
[PI SQL DAS] [PIOLEDBENT] PI System 'pisqldas63' is not
registered.
```

Incorrect PI SQL DAS

```
An error occurred while establishing the connection:
   Type: java.sql.SQLException   Error Code: 0
Message:
  [PI SQL DAS gSOAP Channel] Host not found
Details: get host by name failed in tcp_connect()
```

PI SQL DAS Not Running or Port 5461 Blocked by Access Rule

Note: This error message occurs when the PI SQL DAS port is linked by rules, such as those implemented by McAfee software.

```
An error occurred while establishing the connection:
   Type: java.sql.SQLException   Error Code: 0
Message:
   [PI SQL DAS gSOAP Channel] No connection could be made because
the target machine actively refused it.
```

Firewall Does Not Allow Inbound Connection on PI SQL DAS Side

```
java.sql.SQLException: [PI SQL DAS gSOAP Channel] A connection
attempt failed because the connected party did not properly
respond after a period of time, or established connection failed
because connected host has failed to respond.
Details: connect failed in tcp_connect()
        at
om.osisoft.jdbc.ConnectionImpl.<init>(ConnectionImpl.java:78)
        at com.osisoft.jdbc.Driver.connect(Driver.java:252)
        at java.sql.DriverManager.getConnection(Unknown Source)
        at java.sql.DriverManager.getConnection(Unknown Source)
        at getSnap.main(getSnap.java:22)
```

SSL not configured (error appears after short timeout)

```
An error occurred while establishing the connection:
   Type: java.sql.SQLException   Error Code: 0
Message:
   [PI SQL DAS gSOAP Channel] EOF was observed that violates the
protocol. The client probably provided invalid authentication
information.
Details: SSL connect failed in tcp_connect()
```

Certificate Not Suitable Error Appears Immediately

```
An error occurred while establishing the connection:
   Type: java.sql.SQLException   Error Code: 0
Message:
   [PI SQL DAS gSOAP Channel] EOF was observed that violates the
protocol. The client probably provided invalid authentication
information.
Details: SSL connect failed in tcp_connect()
```

Verify Java Version

In Java, older versions remain installed for compatibility. Programs can explicitly reference an older version or use the general path that should, but do not always, give you the latest version installed. Therefore, depending on the Linux version used, even if you updated the Java version to 1.6 or later, you may still be referencing an older version of Java.

To verify the version and architecture before using it from PI JDBC, enter into a command prompt:

```
administrator@SLES10-32:~> java -version
```

You should see output such as:

```
java version "1.6.0_21"
Java(TM) SE Runtime Environment (build 1.6.0_21-b06)
Java HotSpot(TM) 64-Bit Server VM (build 17.0-b16, mixed mode)
```

If the version does not meet the requirements you may need to redefine the Java link, for example, `/usr/bin/java`, to point to the latest JRE version installed.

OpenSSL Library

Linux

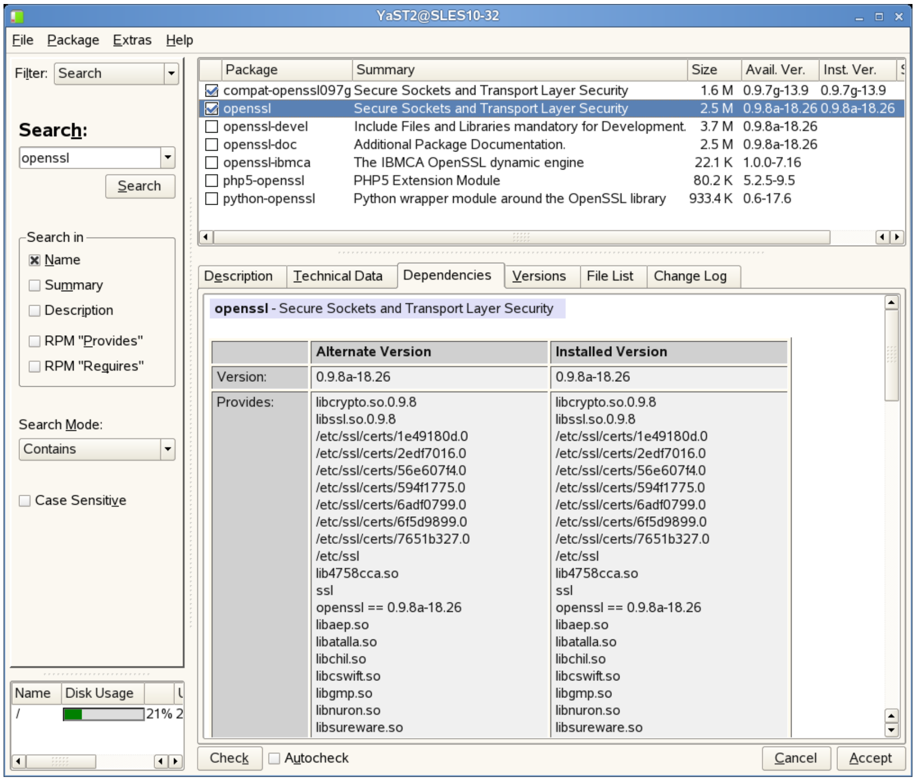
PI JDBC requires an OpenSSL library referenced as `libcrypto.so.and` `libssl.so.`These references are typically links to the real library files that represent a sub-version. Depending on the Linux version, these links may not be defined yet.

1. Verify that these links exist. They are typically located in `/lib` or `/lib64` or `/usr/lib or /usr/lib64.`

PI JDBC supports Linux systems that have or can be updated to OpenSSL 0.9.8 and Linux Systems that come with OpenSSL 1.0.0.

The following table lists examples of Linux systems and what OpenSSL version they are supporting:

Linux	OpenSSL version	Comment
OpenSuse 11.1	0.9.8	
RHEL 5	0.9.8	Needs update for `libstdc++`
Ubuntu 9	0.9.8	
SLES 10	0.9.8	Needs update for `libstdc++`
OpenSuse 11.4	1.0.0	
SLES 11	0.9.8	
RHEL 6	1.0.0	



Filter: Search

Search: openssl

Search in:
☒ Name
☐ Summary
☐ Description
☐ RPM "Provides"
☐ RPM "Requires"

Search Mode: Contains

☐ Case Sensitive

	Package	Summary	Size	Avail. Ver.	Inst. Ver.
☑	compat-openssl097g	Secure Sockets and Transport Layer Security	1.6 M	0.9.7g-13.9	0.9.7g-13.9
☑	openssl	Secure Sockets and Transport Layer Security	2.5 M	0.9.8a-18.26	0.9.8a-18.26
☐	openssl-devel	Include Files and Libraries mandatory for Development.	3.7 M	0.9.8a-18.26	
☐	openssl-doc	Additional Package Documentation.	2.5 M	0.9.8a-18.26	
☐	openssl-ibmca	The IBMCA OpenSSL dynamic engine	22.1 K	1.0.0-7.16	
☐	php5-openssl	PHP5 Extension Module	80.2 K	5.2.5-9.5	
☐	python-openssl	Python wrapper module around the OpenSSL library	933.4 K	0.6-17.6	

openssl - Secure Sockets and Transport Layer Security

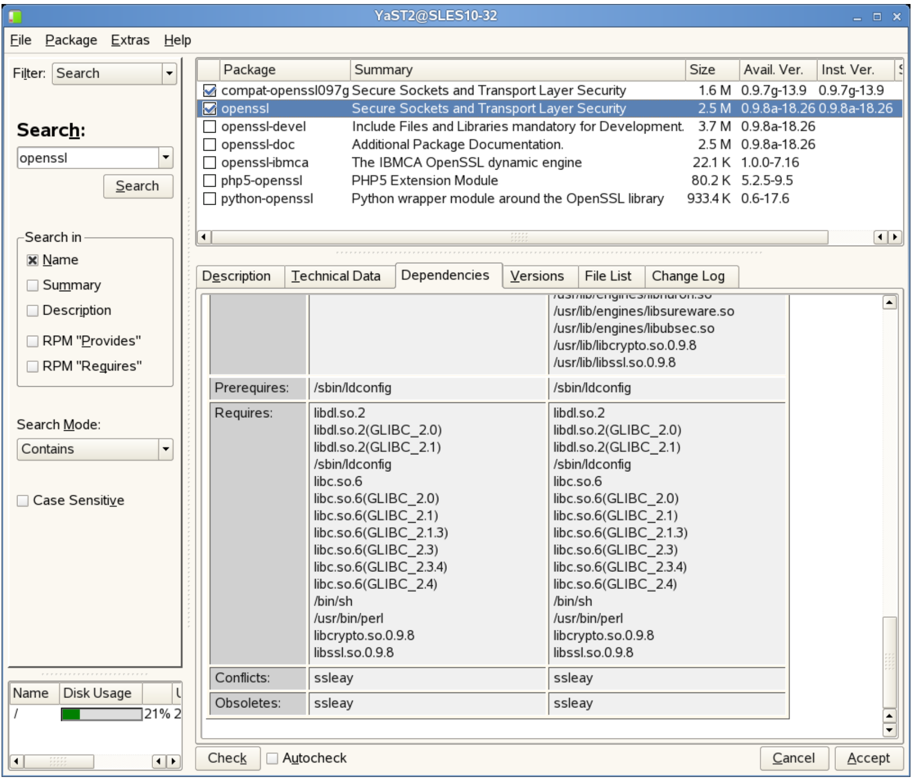
	Alternate Version	Installed Version
Version:	0.9.8a-18.26	0.9.8a-18.26
Provides:	libcrypto.so.0.9.8 libssl.so.0.9.8 /etc/ssl/certs/1e49180d.0 /etc/ssl/certs/2edf7016.0 /etc/ssl/certs/56e607f4.0 /etc/ssl/certs/594f1775.0 /etc/ssl/certs/6adf0799.0 /etc/ssl/certs/6f5d9899.0 /etc/ssl/certs/7651b327.0 /etc/ssl lib4758cca.so ssl openssl == 0.9.8a-18.26 libaep.so libatalla.so libchil.so libcswift.so libgmp.so libnuron.so libsureware.so	libcrypto.so.0.9.8 libssl.so.0.9.8 /etc/ssl/certs/1e49180d.0 /etc/ssl/certs/2edf7016.0 /etc/ssl/certs/56e607f4.0 /etc/ssl/certs/594f1775.0 /etc/ssl/certs/6adf0799.0 /etc/ssl/certs/6f5d9899.0 /etc/ssl/certs/7651b327.0 /etc/ssl lib4758cca.so ssl openssl == 0.9.8a-18.26 libaep.so libatalla.so libchil.so libcswift.so libgmp.so libnuron.so libsureware.so

☐ Autocheck

Note: You may need to install the libraries if they are not installed on the system.

The OpenSSL library is dependent on the `libc` library and requires a certain revision. On some older Linux distributions, such as RHEL5 and SLES10, the libraries must be updated.

1. To verify the presence of `libc.so.6` (GLIBC_2.4) within `glibc`:


YaST2@SLES10-32
File Package Extras Help
Search
Description Technical Data Dependencies Versions File List Change Log
Check
Cancel Accept

Filter: Search

Search: openssl

Search in
☒ Name
☐ Summary
☐ Description
☐ RPM "Provides"
☐ RPM "Requires"

Search Mode: Contains

☐ Case Sensitive

Name	Disk Usage
/	21%

	Package	Summary	Size	Avail. Ver.	Inst. Ver.
☑	compat-openssl097g	Secure Sockets and Transport Layer Security	1.6 M	0.9.7g-13.9	0.9.7g-13.9
☑	openssl	Secure Sockets and Transport Layer Security	2.5 M	0.9.8a-18.26	0.9.8a-18.26
☐	openssl-devel	Include Files and Libraries mandatory for Development.	3.7 M	0.9.8a-18.26	
☐	openssl-doc	Additional Package Documentation.	2.5 M	0.9.8a-18.26	
☐	openssl-ibmca	The IBMCA OpenSSL dynamic engine	22.1 K	1.0.0-7.16	
☐	php5-openssl	PHP5 Extension Module	80.2 K	5.2.5-9.5	
☐	python-openssl	Python wrapper module around the OpenSSL library	933.4 K	0.6-17.6	

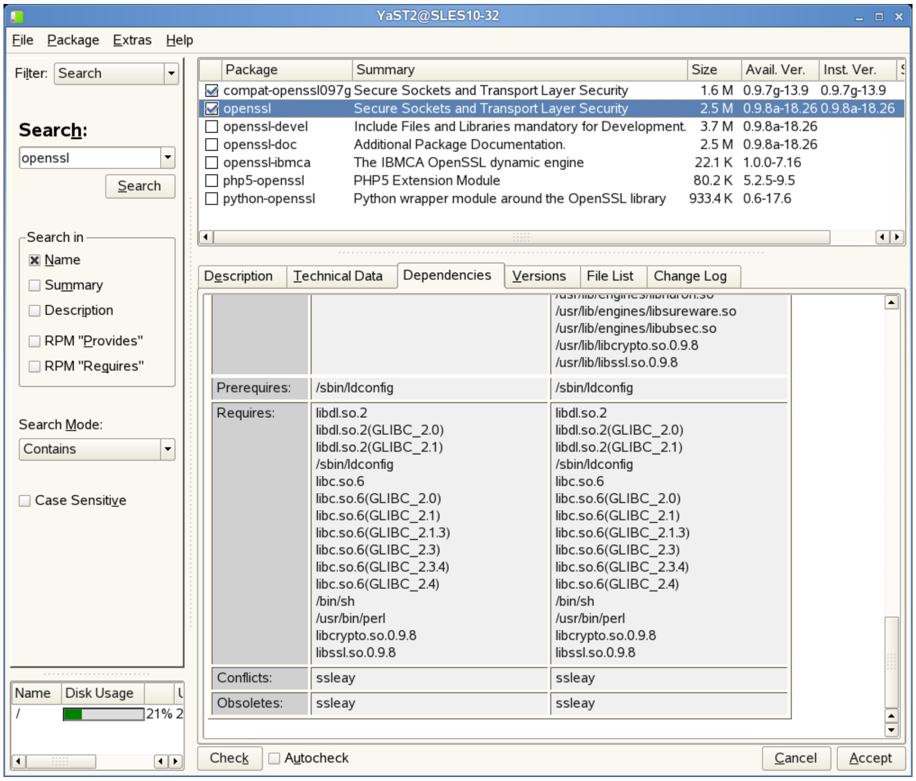
		/usr/lib/engines/libnuron.so /usr/lib/engines/libsureware.so /usr/lib/engines/libubsec.so /usr/lib/libcrypto.so.0.9.8 /usr/lib/libssl.so.0.9.8
Prerequires:	/sbin/ldconfig	/sbin/ldconfig

Requires:	libdl.so.2
	libdl.so.2(GLIBC_2.0)
	libdl.so.2(GLIBC_2.1)
	/sbin/ldconfig
	libc.so.6
	libc.so.6(GLIBC_2.0)
	libc.so.6(GLIBC_2.1)
	libc.so.6(GLIBC_2.1.3)
	libc.so.6(GLIBC_2.3)
	libc.so.6(GLIBC_2.3.4)
	libc.so.6(GLIBC_2.4)
	/bin/sh
	/usr/bin/perl
	libcrypto.so.0.9.8
	libssl.so.0.9.8
		libdl.so.2
		libdl.so.2(GLIBC_2.0)
		libdl.so.2(GLIBC_2.1)
		/sbin/ldconfig
		libc.so.6
		libc.so.6(GLIBC_2.0)
		libc.so.6(GLIBC_2.1)
		libc.so.6(GLIBC_2.1.3)
		libc.so.6(GLIBC_2.3)
		libc.so.6(GLIBC_2.3.4)
		libc.so.6(GLIBC_2.4)
		/bin/sh
		/usr/bin/perl
		libcrypto.so.0.9.8
		libssl.so.0.9.8

Conflicts:	ssleay	ssleay
Obsoletes:	ssleay	ssleay

☐ Autocheck

2. To verify that `glibc` contains libc.so.6(GLIBC_2.4):



Filter: Search

Search: openssl

Search in
☒ Name
□ Summary
□ Description
□ RPM "Provides"
□ RPM "Requires"

Search Mode: Contains

□ Case Sensitive

Name	Disk Usage
/	21%

	Package	Summary	Size	Avail. Ver.	Inst. Ver.
☑	compat-openssl097g	Secure Sockets and Transport Layer Security	1.6 M	0.9.7g-13.9	0.9.7g-13.9
☑	openssl	Secure Sockets and Transport Layer Security	2.5 M	0.9.8a-18.26	0.9.8a-18.26
□	openssl-devel	Include Files and Libraries mandatory for Development.	3.7 M	0.9.8a-18.26	
□	openssl-doc	Additional Package Documentation.	2.5 M	0.9.8a-18.26	
□	openssl-ibmca	The IBMCA OpenSSL dynamic engine	22.1 K	1.0.0-7.16	
□	php5-openssl	PHP5 Extension Module	80.2 K	5.2.5-9.5	
□	python-openssl	Python wrapper module around the OpenSSL library	933.4 K	0.6-17.6	

		/usr/lib/engines/libsureware.so /usr/lib/engines/libubsec.so /usr/lib/libcrypto.so.0.9.8 /usr/lib/libssl.so.0.9.8
Prerequires:	/sbin/ldconfig	/sbin/ldconfig

Requires:	libdl.so.2
	libdl.so.2(GLIBC_2.0)
	libdl.so.2(GLIBC_2.1)
	/sbin/ldconfig
	libc.so.6
	libc.so.6(GLIBC_2.0)
	libc.so.6(GLIBC_2.1)
	libc.so.6(GLIBC_2.1.3)
	libc.so.6(GLIBC_2.3)
	libc.so.6(GLIBC_2.3.4)
	libc.so.6(GLIBC_2.4)
	/bin/sh
	/usr/bin/perl
	libcrypto.so.0.9.8
	libssl.so.0.9.8
		libdl.so.2
		libdl.so.2(GLIBC_2.0)
		libdl.so.2(GLIBC_2.1)
		/sbin/ldconfig
		libc.so.6
		libc.so.6(GLIBC_2.0)
		libc.so.6(GLIBC_2.1)
		libc.so.6(GLIBC_2.1.3)
		libc.so.6(GLIBC_2.3)
		libc.so.6(GLIBC_2.3.4)
		libc.so.6(GLIBC_2.4)
		/bin/sh
		/usr/bin/perl
		libcrypto.so.0.9.8
		libssl.so.0.9.8

Conflicts:	ssleay	ssleay
Obsoletes:	ssleay	ssleay

Note: RPM packages that contain the necessary files typically use the naming convention `libstdc++…4.2.1…rpm`. For example, `libstdc++42-4.2.1_20070604-1.x86_64.rpm`, which can be used for RHEL5 (x64).

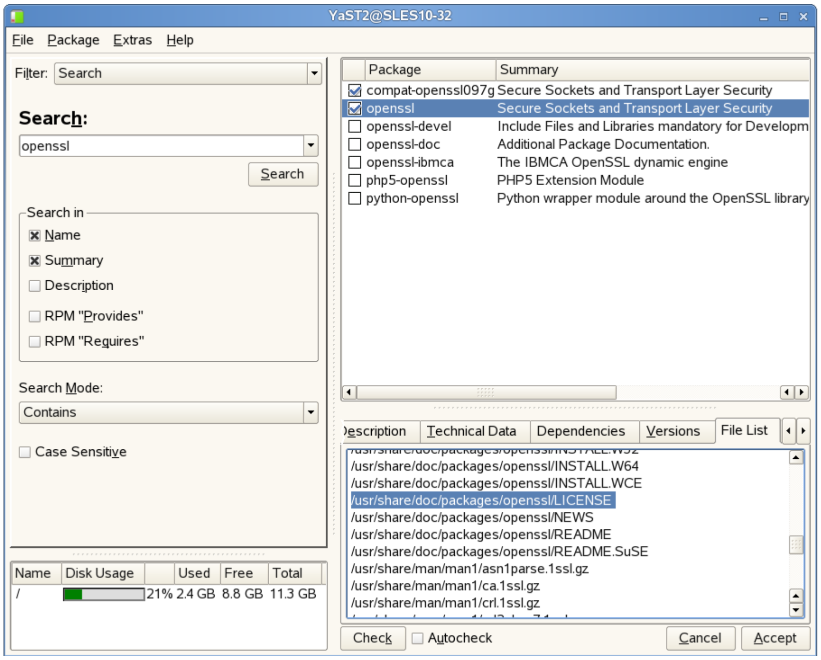
Windows

The OpenSSL library is included in the PI JDBC driver.

OpenSSL License

On Linux, PI JDBC dynamically links the OpenSSL library files that are installed as part of the Linux distribution. The related license information is part of the Linux distribution.

For example, in SLES10, the license information is found in the **File List**:



Filter: Search

Search: openssl

☒ Name
☒ Summary
☐ Description
☐ RPM "Provides"
☐ RPM "Requires"

Search Mode: Contains

☐ Case Sensitive

Name	Disk Usage	Used	Free	Total
/	21%	2.4 GB	8.8 GB	11.3 GB

	Package	Summary
☑	compat-openssl097g	Secure Sockets and Transport Layer Security
☑	openssl	Secure Sockets and Transport Layer Security
☐	openssl-devel	Include Files and Libraries mandatory for Developm
☐	openssl-doc	Additional Package Documentation.
☐	openssl-ibmca	The IBMCA OpenSSL dynamic engine
☐	php5-openssl	PHP5 Extension Module
☐	python-openssl	Python wrapper module around the OpenSSL library

```
/usr/share/doc/packages/openssl/INSTALL.W32
/usr/share/doc/packages/openssl/INSTALL.W64
/usr/share/doc/packages/openssl/INSTALL.WCE
/usr/share/doc/packages/openssl/LICENSE
/usr/share/doc/packages/openssl/NEWS
/usr/share/doc/packages/openssl/README
/usr/share/doc/packages/openssl/README.SuSE
/usr/share/man/man1/asn1parse.1ssl.gz
/usr/share/man/man1/ca.1ssl.gz
/usr/share/man/man1/crl.1ssl.gz
```

On Windows, PI JDBC includes the OpenSSL libraries. Please see the PI JDBC Release Notes for the location of the license and copyright information document.

Appendix A

Tested PI JDBC Scenarios

PI JDBC Linux (x86) and PI SQL DAS (x86)
PI JDBC Linux (x86) and PI SQL DAS (x64)
PI JDBC Linux (x64) and PI SQL DAS (x86)
PI JDBC Linux (x64) and PI SQL DAS (x64)
PI JDBC Windows (x86) and PI SQL DAS (x86)
PI JDBC Windows (x86) and PI SQL DAS (x64)
PI JDBC Windows (x64) and PI SQL DAS (x86)
PI JDBC Windows (x64) and PI SQL DAS (x64)
eclipse Indigo Service Release 1
DBVisualizer 7.1.3 and 8.0.1 (Free and Personal editions)
Java SE Runtime Environment from 1.6.0_02-b06 to 1.6.0_21-b06
Java SE Runtime Environment 1.7.0-b147
SQL Developer 2.3.0
SQuirreL SQL Client Version 3.4.0
Tomcat 6.0

Appendix B

Technical Support and Resources

For technical assistance, contact OSIsoft Technical Support at +1 510-297-5828 or techsupport@osisoft.com. The *OSIsoft Technical Support* (*http://techsupport.osisoft.com*) website offers additional *contact options* (*http://techsupport.osisoft.com/Contact+Us/Contact+Methods/Contact+Methods.htm*) for customers outside of the United States.

When you contact OSIsoft Technical Support, be prepared to provide this information:

- Product name, version, and build numbers
- Computer platform (CPU type, operating system, and version number)
- Time that the difficulty started
- Log files at that time
- Details of any environment changes prior to the start of the issue
- Summary of the issue, including any relevant log files during the time the issue occurred

The *OSIsoft Virtual Campus (vCampus)* (*http://vCampus.osisoft.com*) website has subscription-based resources to help you with the programming and integration of OSIsoft products.

Index

A

About the OSIsoft PI Data Access Suite • 2
Add the Driver to the CLASSPATH Variable • 20
Advanced JDBC • 24
Architecture • 2

B

Batch Processing • 24

C

Certificate Not Suitable Error Appears Immediately • 34
Configuration • 21
Configuration Tool • 8
Configure PI SQL Data Access Server (optional) • 8
Connect with PI SQL Data Access Server • 16
Connection Messages and Errors • 31

D

Data Types • 29
DBVisualizer • 27
Deployment Options • 3
Driver Class • 21
Driver Properties • 22

F

Files • 7
Files Installed • 13, 18
Firewall Does Not Allow Inbound Connection on PI SQL DAS Side • 33
Functionality • 24

G

getEASnap • 15, 20
getSnap • 14

I

In this Guide • 1
Incorrect Data Source (Server) Name • 32
Incorrect PI SQL DAS • 33
Incorrect Username/Password for Connection to PI SQL DAS • 32
Install PI JDBC Driver on Linux • 17
Install PI JDBC Driver on Windows • 12
Install PI SQL Data Access Server • 6
Installation • 5
Installation Components • 5
Installation Result • 7
Installation Results • 14
Introduction • 1

J

JDBC Driver Extensions • 24

L

Linux • 34

M

Message Logging • 25
Middleware Deployment • 4
Multiple Connections • 29
Multithreading • 24

N

No SSPI Configuration or Incorrect Username/Password • 32

O

OpenSSL Library • 34
OpenSSL License • 38

P

PI Initialization Properties • 29
PI JDBC • 25
PI JDBC Driver Limitations • 29
PI Server Login • 16
PI Server Login through PI SQL DAS • 10
PI SQL DAS • 25
PI SQL DAS Not Running or Port 5461 Blocked by Access Rule • 33

R

Remove PI JDBC • 17
Remove PI JDBC (Linux) • 20
Remove PI SQL DAS • 12
Run PI SQL DAS Interactively • 31
Run the PI JDBC Driver Setup Kit • 12

Run the PI SQL DAS Setup Kit • 6

S

Security • 16
Set up PI JDBC Driver on Linux • 17
Source Code Example • 26
SQL • 24
SSL not configured (error appears after short timeout) • 33
SSL Port • 10
Standalone Deployment • 4
Successful Connection • 32
System Requirements • 5
System Variable PI_RDSA_LIB • 19

T

Technical Support and Resources • 41
Tested PI JDBC Scenarios • 39
Time Zone • 21, 29
Transactions • 29
Troubleshooting • 31

U

URL Format • 21
Use Cases • 26
Using PI JDBC • 21

V

Validating Installation • 20
Verify Java Version • 34
Verify the PI JDBC Installation • 14

W

Windows • 37