
# *PegaRULES Process Commander Release V6.1 SP2*

## *Installing on Tomcat Application Servers*

September 25, 2010

**© Copyright 2010**
**Pegasystems Inc., Cambridge, MA**
All rights reserved.

This document and the software describe products and services of Pegasystems Inc. It may contain trade secrets and proprietary information. This information should not be disclosed to third parties unless otherwise provided for by a contract or business agreement with Pegasystems. This document and the software are protected by federal copyright law, international laws, and/or applicable treaties.

This document is current as of the date of publication only. Changes in the document may be made from time to time at the discretion of Pegasystems. This document remains the property of Pegasystems and must be returned to it upon request. This document does not imply any commitment to offer or deliver the products or services provided.

This document may include references to Pegasystems product features that have not been licensed by your company. If you have questions about whether a particular capability is included in your installation, please consult your Pegasystems service consultant.

Other brand or product names are trademarks of their respective holders.

This document is the property of:
Pegasystems Inc.
101 Main Street
Cambridge, MA 02142-1590

Phone: (617) 374-9600
Fax: (617) 374-9620
www.pega.com

PegaRULES Process Commander
Document: Installation Guide
Software Version: 6.1 SP2
Updated: September 25, 2010

# Contents

# Introduction

This document describes how to install a new instance of PegaRULES Process Commander 6.1 SP2. Upgrading an existing Process Commander installation is described in other documents:

- To upgrade a Process Commander 4.2 system to 6.1 SP2, your current system must first be at 4.2 SP6. If your current system is earlier than 4.2 SP6, see the Upgrade Guide for 4.2 SP6 for instructions on moving to SP6.
- To upgrade from 4.2 SP6 to 6.1 SP2, see the manual Upgrade Guide, V4.2 SP6 to V5.
- For information about upgrading any existing Process Commander 5 installation to 6.1 SP2 see the PegaRULES Process Commander Upgrade Guide 6.1 SP2, released with Process Commander 6.1 SP2.

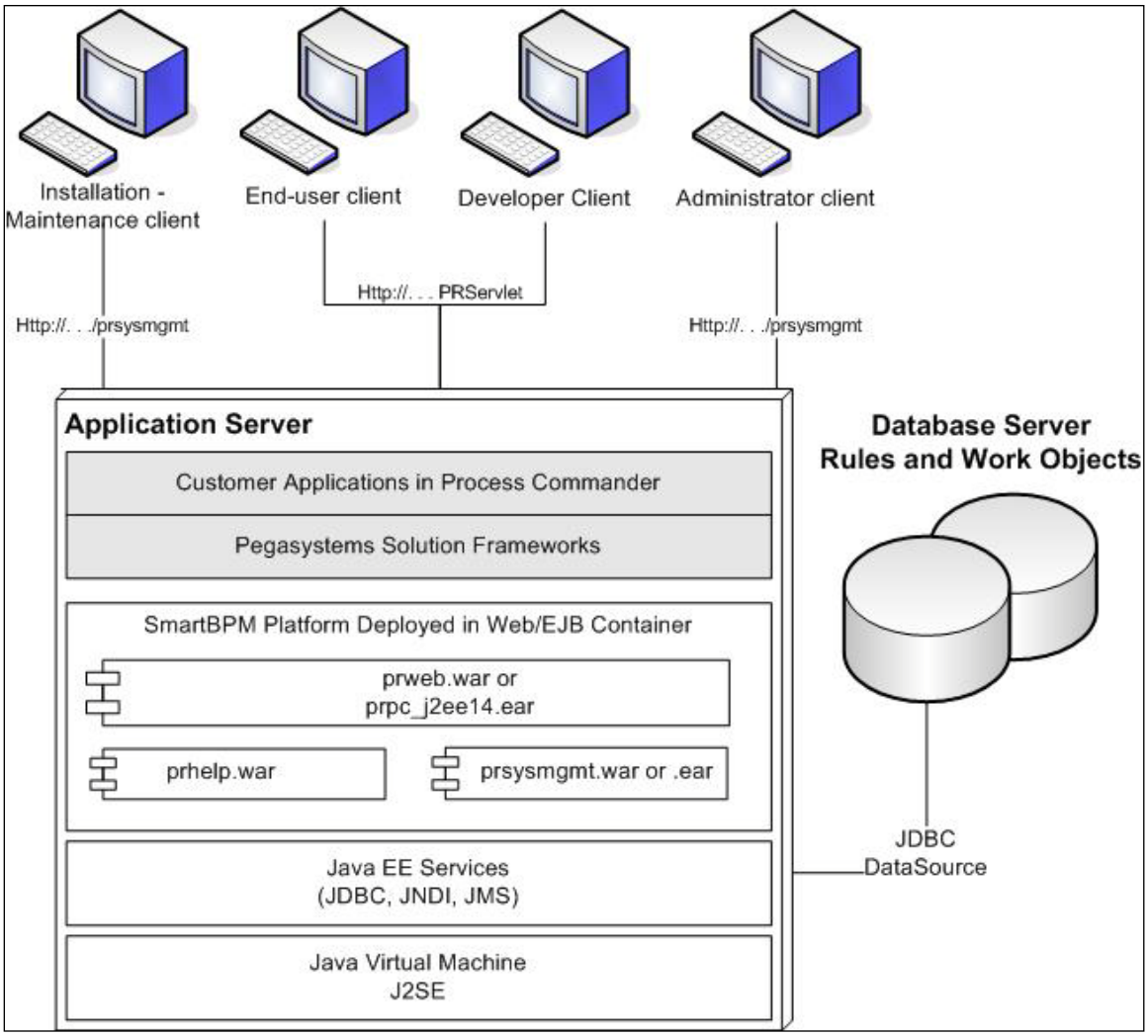
These documents and any updates to existing Process Commander documentation are available on the Pega Developer Network (pdn.pega.com). Please check this site for the latest installation and configuration information.

## Process Commander Deployment Architecture

Process Commander is a Java EE enterprise application. As shown in the diagram below, it runs on top of two main platform sub-systems:

1. A database system to store the rules and work objects used and generated by Process Commander.

2. An application server that supports the Java EE specification. The application server provides a run-time environment and other services, such as database connections, Java Messaging Services (JMS) support, and connector and services interfaces to other external systems.




Both developers working on Process Commander applications and users working with those applications access Process Commander through Web browser interfaces.

# Overview of the Installation Procedure

The following outline details the steps in the 6.1 SP2 installation process:

| | |
|---|---|
| Preparation Phase | Step 1 – Environment Considerations |
| | Step 2 – Database Preparation |
| | Step 3 – Application Server Preparation |
| Rulebase Setup Phase | Step 4 - License Agreement |
| | Step 5 - Type of Setup |
| | Step 6 - Database Configuration |
| | Step 7 - Database Connection |
| | Step 8 - Temporary Directory |
| | Step 9 - Rulebase Loading |
| PRPC Configuration Phase | Step 10 – Final Settings in PRPC |

Installing Process Commander includes 10 steps organized into three phases:

## 1. Preparation Phase

You complete the **Preparation Phase** following the procedures in this guide.

- **Step 1: Environment Considerations**, page 6 .

Make sure that your environment meets the requirements for PRPC 6.1 SP2 in terms of: Hardware, OS, JVM, JDBC Driver, Application Server, Database Server, PRPC, and their respective settings.

1. Review the current Platform Support Guide, which is available on the Pega Developer Network (PDN) or from your Pega account executive. This document lists the recommended software components needed to support Process Commander, and any special requirements or known issues with particular vendors or versions.
2. Review your site's deployment requirements with your IT representatives.
3. Consult with your Pegasystems representative about your installation plans. Pegasystems' prior experience with various platforms and deployment architectures can help your installation go smoothly.

- **Step 2: Database Preparation**, page 9.

Create and configure a blank database for Process Commander to use. You can use SQL scripts provided by Pegasystems to create the schema required by Process Commander in the database, or you can choose to have the Installation and Upgrade Assistant create the schema when you run that tool.

You provide connection information for this database to the Installation and Upgrade Assistant used in the Rulebase Setup Phase and to specify the datasource in your application server. The connection information you will need includes the location of the JDBC driver file, the driver class name, the database connection URL, and the database user's username and password.

Note: When the Database Preparation is complete, Step 3, the Application Server Preparation, and Steps 4-9 in the Rulebase Setup Phase can be performed in either order or simultaneously. You can, for example, launch the Rulebase Setup process in Step 9 of the Assistant and then configure your Application Server while the Rulebase load process is running.
However, you must configure the application server and deploy the Process Commander applications before you can complete Step 10, PRPC Final Settings.

Important: You must be careful not to start the deployed Process Commander applications while the Rulebase loading process is still running on the database. Note that your application server may start the application automatically when it is deployed. If the Process Commander application attempts to connect to the database while the Rulebase is loading, the database loading will not complete correctly.

- **Step 3: Application Server Preparation ,** page 14.

Configure your application server to support the Process Commander application, including:

1. Configure the right Database Driver for your Database
2. Configure the PRPC Data Source
3. Configure other Application Server settings
4. Deploy the Process Commander application (WAR or EAR)
5. Deploy the online help application (prhelp.war) and System Management Application (prsysmgmt.war).

## 2. Rulebase Setup Phase

You complete the **Rulebase Setup Phase** by running steps 4 to 9 in the Installation and Upgrade Assistant (PRPC_Setup.jar). This tool is covered in this guide in the section Steps 4-9, Rulebase Setup Phase on page 31 .

The Installation Assistant optionally creates the Process Commander schema in the database instance you provide, and installs the initial set of rules and classes Process Commander needs to run into the database.

### 3. PRPC Configuration Phase

You complete the **PRPC Configuration Phase** after finishing the Installation by logging into Process Commander. The setup steps you should complete in Process Commander are described in section Step 10: Final PRPC Configuration Steps, page 46.

## Process Commander Core Component Versions

The Process Commander core application is packaged in four different versions to support deployment into different Java EE environments. The following table summarizes which packages provided and their use:

| Type of deployment | PRPC package to use |
|---|---|
| All Tomcat deployments | prweb.war |
| New deployments on WebSphere 6.1.x, 7.0.x. | prweb.war or prpc_j2ee14_ws.ear |
| New deployments on WebLogic 9.x, 10.3.1, 11g (10.3.2); JBoss CE 4.2.x, 4.3.x, 5.0.x | prweb.war or prpc_j2ee14.ear |
| New deployments on JBoss EAP 4.3.x, 5.0.x (Red Hat) | prpc_j2ee14_jbossJBM.ear |

Tomcat users should install the WAR in all cases.

If you have any questions regarding the correct package to deploy, contact Pega Support.

# Step 1:
# Environment Considerations

## Prerequisites

The prerequisites for installing Process Commander are:

- An application server host computer with adequate memory and disk space for the installation.

  Process Commander requires a minimum of one GB of free memory. This is in addition to any memory allocated for the operating system or other applications on the server. The memory required for your system may need to be higher, depending on the number of users and your specific configuration.

- A working installation of the application server.

  Process Commander Installation Guides for each supported application server are available with your Pegasystems software distribution and on the Deployment page of the Pega Developer Network (http://pdn.pega.com).

- A configured relational database available for Process Commander use, and support in your organization for the installation and configuration of relational databases.

The database and application server platforms may be either UNIX or Windows systems. However, a Windows system is required for the Pegasystems developer and end-user clients and for some of the Pegasystems monitoring tools, specifically, PegaRULES® Process Analyzer™, Process Simulator™. The

environment for your initial Process Commander system should include at least one Windows platform.

> Note: If you have difficulty installing any of the third-party software required by Process Commander, contact the other software vendor to resolve the issues before installing Process Commander.

## Supported Process Commander Platforms

Because Process Commander is developed to industry standards — Java and JEE technologies, JEE compliant application servers, SQL databases, web browsers, XML, and HTML— a Process Commander system can be successfully deployed on a wide range of hardware and software platform combinations. However, careful selection of your platform components can make your installation and deployment of Process Commander easier.

To determine the right platform components for your Process Commander deployment:

1. Review the current *Platform Support Guide,* which is available on the Pega Developer Network (PDN) or from your Pega account executive. This document lists the recommended software components needed to support Process Commander, and any special requirements or known issues with particular vendors or versions.
2. Review your site's deployment requirements with your IT representatives.
3. Consult with your Pegasystems representative about your installation plans. Pegasystems' prior experience with various platforms and deployment architectures can help your installation go smoothly.

For additional information see the Platform Support Guide, which is available as a PDF file on the Pega Developer Network (PDN), located at http://pdn.pega.com.

## Check the Pega Developer Network for the Latest Information

Before beginning your installation procedure, check the Pega Developer Network (PDN), located at http://pdn.pega.com, for the latest information. The installation guides are updated whenever new information becomes available. Be sure to check for the latest version.

### Pegasystems Documentation

The latest Process Commander documentation, including Installation, Upgrade, and Platform Support guides is available on the Documentation page of the Pega Developer Network.

Visit http://pdn.pega.com and select **Guides -> Process Commander V6 Deployment**

### Deployment information and updates

Check the Deployment page on the Pega Developer Network for information, updates, and platform bulletins on deploying Process Commander including links to the latest Knowledge Base articles.

Visit http://pdn.pega.com and select **Browse -> Deployment -> About This Topic**

### Pegasystems Training

A range of Pegasystems course offerings are available to increase the effectiveness of your deployment and development team members. For more information, visit http://www.pega.com/Services/EducationalServices

## A Note for PRPC 4.2 Users

This installation procedure differs from earlier installation procedures in significant ways. Please review the instructions carefully. In particular, please be aware of two changes:

- Pegasystems now recommends using data sources defined in your application server to establish connections to your database instead of the Process Commander connection pooling sometimes used in earlier installations.
- Be sure to create and configure a temporary directory that Process Commander can use to store static data. See the section “Setting the Explicit Temporary Directory” in the chapter that describes deployment with your application server.

# Step 2:

# Database Preparation

To prepare your database server to use with Process Commander, you must complete the following tasks:

- Create a blank database.
- Create an appropriate user that Process Commander can use to access the database.
- Create a database schema that defines the data objects (tables, indexes, views, triggers and procedures) required by Process Commander.

  You can either have your DBA, or someone with similar permissions, use your database tools to run a SQL script provided by Pegasystems to create the schema objects, or, in Step 6 of the Assistant, you can select a Create Schema option to have the Assistant create the schema automatically before loading the Rulebase.

  Running the SQL script allows you to review and, if necessary, customize the schema for your environment.

## Database Prerequisites

Process Commander requires a working installation of one of the supported databases, as described below.

For additional information, see the Platform Support Guide, which is available as a PDF file on the Pega Developer Network (PDN). Go to http://pdn.pega.com and select **Guides -> Process Commander V6 Deployment**.

Install and configure the relational database according to vendor directions. This database may reside on the same server as the Process Commander software, or it may be on a different server.

# Creating a PRPC Database

Create a database with a minimum of 4GB for the user tablespace and allow the database to grow. For a typical development environment, expect the database to grow to at least 10GB over time. The space required for your system may need to be higher, depending on your use of Process Commander and the number of users.

Be sure to record:

- Database server name
- Database port number

The Process Commander system requires this information to access this database.

## Additional UDB Requirement

When using UDB, for each schema that contains Process Commander tables, Process Commander requires a 32K bufferpool, a 32K user regular tablespace, and a 32K user temporary tablespace. By default, the UDB database create process creates a bufferpool and temporary tablespace with only a 4K page size.

# Creating Access to the Database

Create a user for this database with full read and write permissions. At a minimum this “base” Process Commander user must have appropriate read and write permissions to be able to run SQL scripts on the database. In addition, a small number of Process Commander functions require permission to run CREATE_TABLE and ALTER_TABLE operations. You can assign permissions that include these operations to the base user, or you can create a second Admin DB user with these extended permissions which Process Commander uses only for these special functions. For details, see “Creating Base and Admin Database Users,” below.

When you configure data access in your application server, you must provide the user name and password for the user or users you have created to access the Process Commander database. Be sure to record this information and have it available when configuring the application server.

## Creating Base and Admin Database Users

If possible, it is best to create a single Process Commander user with full permissions. If you want to restrict users’ access to the Process Commander database, you can create two users:

- A “base” user with only read and write permissions.
- An Admin user with full access including permission for CREATE_TABLE and ALTER_TABLE operations.

To implement two users, you configure access to the database in your application server by creating two data sources:

- One datasource referencing the JNDI location **jdbc/PegaRULES** that specifies the username and password for the base DB user. This user must have at least read and write permissions for the database.
- A second datasource referencing the JNDI location **jdbc/AdminPegaRULES** specifying the username and password for the DB Admin user. This user must have full access including permission for CREATE_TABLE and ALTER_TABLE operations.

### How Process Commander Uses the two Database Accounts

If you configure a second datasource for the Admin user, Process Commander uses that user when performing a small number of functions that require altering the database, specifically:

- Using the Database Schema Management tool (Tools > Database > Modify Database Schema).
- Creating any new Circumstance Definition rule or defining a single circumstance on a rule. The extended permissions are needed to support reporting on circumstanced properties. (On the Pega Developer Network, see PRKB-25173: How to create a rule with multiple circumstance properties.)
- Exposing embedded properties for selection and reporting. (On the Pega Developer Network, see PRKB-25278: How to create Declarative indexes for embedded properties.)

For all other functions, Process Commander uses the base DB user which only requires read and write permissions.

If you do not provide ALTER TABLE and CREATE TABLE permissions to any user, Process Commander generates a warning when you use these features. The warning includes SQL script that you can provide to your DBA to manually make the necessary database changes.

Follow the guidelines for the appropriate database in the following sections

## User Permissions for the Microsoft SQL Database

Set the user account for the Process Commander database user with **public** and **db_owner** roles with all corresponding System Permissions granted.

## User Privileges for the Oracle Database

Set the user account for the Process Commander database user with the default RESOURCE and CONNECT roles with all corresponding system privileges granted. In particular, be sure that the user has the Create View system privilege.

> Note: On the Oracle 10g HTML-based Enterprise Manager console there are separate tabs for setting the user's roles and system privileges. When editing the properties for the Pega user, be sure to go to the Roles tab to set the RESOURCE and CONNECT roles, and then to the System Privileges tab to set the Create View system privilege.
>
> Important: Do not grant DBA access to the user, as this may cause problems with the installation.

## User Privileges for the UDB Database

Create a user account for the Process Commander database user with at least SYSADM privileges.

# Create the Schema in Process Commander Database

Before you load the initial Process Commander rules into the database in Steps 4-9 below, you must create the database schema that defines the data objects (tables, indexes, views, triggers and procedures) required by Process Commander.

You can create the schema in two ways:

1. In the Install and Upgrade Assistant user interface, you can select the Create Schema option. The Assistant then creates a default schema in the database automatically before loading the Rulebase.

   If you want the Assistant to create the schema, skip the following section "Using a SQL Script to Create the Schema", and continue with the Assistant as described in Steps 4-9 below.

2. Your database administrator (DBA), or someone with appropriate permissions, can run a SQL script, provided by Pegasystems, to create the schema in your Process Commander database. Process Commander V6.1 SP2 includes SQL scripts for each supported database: mssql, oracle (oracledate), and udb, for example, **oracledate_install_6.1SP2.sql**.

   Running the SQL script allows you to review and, if necessary, customize the schema for your environment.

   To create the schema in the database using the SQL script, follow the procedure described in the next section.

## Using a SQL Script to Create the Schema

If you choose not to select the **Create Schema** option in the Installation and Upgrade Assistant your database administrator (DBA), or someone with appropriate permissions, must run a SQL script to create the schema in your Process Commander database.

1. In the software distribution directory for Process Commander find the install SQL script for your database type.

```
C:\PRPC_distro\schema\oracledate\install\oracledate_install_6.1SP2.sql
```

2. You or your DBA should review this script to make sure it is appropriate for your environment. You can customize the script if necessary. For example, the default scripts create an ASCII database. Modify the scripts if you require Unicode.

   In most cases, the SQL script can be used without modification. If you are unsure about whether the script can be run in your environment, contact Pegasystems Customer Support.

3. Use your database tools to run the appropriate SQL script against your Process Commander database.

> Note: For MS-SQL installations only, the following message may appear:
>
> Warning: The table 'pr4_rule' has been created but its maximum row size (8506) exceeds the maximum number of bytes per row (8060). INSERT or UPDATE of a row in this table will fail if the resulting row length exceeds 8060 bytes.
>
> You may ignore this message. This warning message relates to a known limitation of Microsoft SQL Server. Process Commander does not exceed the maximum row size in this table.

# Next Steps

You can continue either with Step3, Application Server Configuration, or Steps 4-9, Rulebase Setup. These two procedures are independent and can be completed in either order or simultaneously. Note, however, that you cannot run Process Commander successfully until the database setup is complete. If you deploy and start the Process Commander application before the database is complete, the application generates errors.

# Step 3:
# Preparing the Application Server

This chapter describes configuration tasks you must complete on your application server before beginning to deploy the PRPC application, and the procedures for deploying the Process Commander applications.

To prepare the application server before deploying the applications, you:

- Verify the Java Environment
- Install Database Drivers on your Application Server
- Configure Tomcat JVM and other settings
- Configure Data Source Connection and ExplicitTempDir

To deploy the Process Commander system, you deploy three applications:

- the Process Commander core application, prweb.war or prpc_j2ee14.ear
- the System Management Application, prsysmgmt.war or prsysmgmt.ear
- the Developer Help application, prhelp.war

## Verify the Java Environment

Confirm that the Java JDK environment used by your application server is Java JDK 1.5.0_07 or higher.

Refer to the Platform Support Guide, available as a PDF file on the Pega Developer Network (PDN) located at http://pdn.pega.com, for specific recommendations for your platform.

> Important: During installation, Process Commander requires access to the directory specified by the java.io.tmpdir setting in the JVM. Confirm that there is write access to this directory before installing Process Commander.

Be sure your application server is configured to use an appropriate JDK. The default Java environment for your application server may not be one of the recommended JDKs for Process Commander.

## Install Database Drivers on your Application Server

In order to support JDBC connections between your application server and the Process Commander, you must install the appropriate database client software and JDBC drivers on your application server as recommended by your database vendor.

### Microsoft SQL Server

Use the drivers provided by your application server or download the latest drivers from the Microsoft website.

For both SQL Server 2005 and 2008, use the appropriate driver for your JVM

- For Java 5: **sqljdbc.jar**
- For Java 6: **sqljdbc4.jar**

Install the drivers into the <*Tomcat_HOME*>**/common/lib** directory for Tomcat 5 installations, or <*Tomcat_HOME*> **/lib** for Tomcat 6 installations.

### Oracle

For Process Commander to work with the application server using Oracle, follow all Oracle instructions for installing the client software and the JDBC drivers.

1. Confirm that the appropriate JDBC driver is in the **/jdbc/lib** directory of your Oracle installation directory or download it from the Oracle site.

   The appropriate file depends on the Java version of your JVM:

   - For Java 5: **ojdbc5.jar**
   - For Java 6: **ojdbc6.jar**

2. Copy the driver file into the <*Tomcat_HOME*>**\common\lib** directory for Tomcat 5 installations, or <*Tomcat_HOME*> **\lib** for Tomcat 6 installations..

3. If you are using the OCI (thick) driver, confirm or create the following environment variables for your system.

> Note: Pegasystems recommends use of the Thin Client driver.

**Solaris**:

- Set ORACLE_HOME and LD_LIBRARY_PATH environment variables.
- Add $ORACLE_HOME/lib32 to the LD_LIBRARY_PATH *before* the reference to the standard $ORACLE_HOME/lib directory (as Process Commander uses 32-bit libraries).

**AIX**:

- Set ORACLE_HOME and LIBPATH environment variables.
- Add $ORACLE_HOME/lib32 to the LIBPATH *before* the reference to the standard $ORACLE_HOME/lib directory (as Process Commander uses 32-bit libraries).

**Windows:**

On Windows, the necessary environment variables are usually set by the Oracle installation program. Confirm that the ORACLE_HOME is set to your database installation directory and that the Path environment variable includes the path to the Oracle bin directory.

## UDB

For Process Commander to work with a web application server using DB2 UDB, follow the IBM instructions for installing the client software and the JDBC drivers. For DB2, the JDBC driver **db2jcc.jar** file is required. This JDBC driver is Type 4.

For Tomcat, use the DB2 installation procedure to install the driver into the <*Tomcat_HOME*>**/common/lib** directory for Tomcat 5 installations, or <*Tomcat_HOME*> **/lib** for Tomcat 6 installations.

## Setting JAVA_HOME

Confirm that the application server has a JAVA_HOME environment variable set in your environment; it must be configured to point to the top level of your current Java installation. If this variable is not set properly, the following error results when the application server attempts to start:

“Cannot find the file ‘-Xms128m’ (or one of its components). Make sure the path and filename are correct and that all required libraries are available.”

To check the JAVA_HOME environment variable on Windows, select ***Start > Settings > Control Panel > System > Advanced > System Variables***.

# Configuring Tomcat Before Deploying PRPC

Make the following changes to configure Tomcat for the Process Commander installation. Complete instructions are given in the following sections.:

- Setting JVM Memory Parameters
- Enable AWT to support PRPC Graphical Reports
- Setting User ID and Password in Tomcat
- Enabling DNS Lookup
- Setting Localization Libraries
- Create a Data Source

> Note: Be sure to install the appropriate JDBC drivers, as described in Install Database Drivers on your Application Server on page 15.

After entering these changes into the Tomcat system, stop and restart the Tomcat server.

For information on customizing Tomcat, see the Process Commander Support Knowledgebase. Tomcat documents will also be installed in the Tomcat installation, under the *TomcatInstallDirectory*\webapps\tomcat-docs subdirectory.

## Setting JVM Memory Parameters

You must set the JVM memory options to increase the amount of system memory allocated to the server running PRPC. If the server does not have enough memory allocated to run Process Commander, the system may hang without an error message. The following *minimum* values are recommended:

- Initial Java heap size (Xms): 768M
- Maximum heap size (Xmx): 1024M

> Note: These values are initial minimum estimates for typical development systems. They may need to be higher based on your server hardware and the number of other applications on the server. Consult with your System Administrator about the requirements for your site.
>
> Setting the Initial and Maximum heap sizes to the same value helps to avoid the performance costs of heap contraction and expansion. All of the memory is allocated at startup. However, if you set an initial heap size lower than the

maximum, you may find that the heap does not grow to the maximum size. Consult with your System Administrator to tune these settings to the best values for your environment.

If your application server is using the Sun JVM, also add the following PermSize and MaxPermSize settings:

- PermSize: 64MB
- MaxPermSize: 256MB

The method you use to set the memory parameters for the JVM depends on the way you installed Tomcat:

- If you installed Tomcat using the .zip file, on Windows or UNIX, and are not running Tomcat as a service on Windows, create a setenv .bat or .sh file containing the settings. See Creating the setenv File below.
- If you installed Tomcat on Windows using the .exe installation program, Tomcat is automatically installed as a service. You must set the JVM options for the Tomcat service using a command entered at a command-prompt window. See "Setting the Memory Parameters for the Windows Service" on page 19.
- If you install Tomcat on Windows using the .zip file and then install it as a service using the service.bat file distributed with .zip distribution, you must include the memory settings in the service.bat file before running it. See Setting the Memory Parameters in the service.bat File on page 19.

## Creating the setenv File

Use this method to set the memory parameters on UNIX systems, or if you do not run Tomcat as service on Windows.

Create a startup script, **setenv.bat** for Windows or **setenv.sh** for UNIX, in the ***TomcatInstallDirectory*\bin** directory. These scripts do not exist in a default installation.

If Tomcat is installed on a Windows machine, create the **setenv.bat** file with the following information:

```
rem Change -Xmx value based on available physical memory
set JAVA_OPTS=-Xms768m –Xmx1024m –XX:PermSize=64m –XX:MaxPermSize=256m
rem End of lines added by PegaRULES Installation Process
```

For a UNIX server, create a **setenv.sh** file:

```
# Change -Xmx value based on available physical memory
JAVA_OPTS="-Xms768m –Xmx1024m –XX:PermSize=64m –XX:MaxPermSize=256m"
```

```
# End of lines added by PegaRULES Installation Process
```

Note: The JAVA_OPTS settings should all be on one line.

### Setting the Memory Parameters for the Windows Service

If you used the Windows installation program, you must set the JVM options for the Tomcat service using the following command.

Open a command prompt window and change directories to the **bin** directory of your Tomcat installation. At the command prompt, enter the following command:

```
.\tomcat5.exe //US//Tomcat5 ++JvmOptions=-Xms768m;
-Xmx1024m;-XX:PermSize=64m;-XX:MaxPermSize=256m
```

or

```
.\tomcat6.exe //US//Tomcat6 ++JvmOptions=-Xms768m;
-Xmx1024m;-XX:PermSize=64m;-XX:MaxPermSize=256m
```

This command permanently sets the JVM options for the Tomcat service. These memory settings will be used when you start the Tomcat service.

### Setting the Memory Parameters in the service.bat File

The .zip distribution of Tomcat includes a file, service.bat, that you can use to install and uninstall Tomcat as a service on Windows. You can find this file in the bin directory of your Tomcat installation. If you use this file to install Tomcat as a service, include the memory parameters, as described above, as JVM options in the file. The file contains instructions and examples for setting JVM options.

## Enable AWT to support PRPC Graphical Reports

You must add a “headless” AWT setting to the JAVA options in the setenv.bat (Windows) or the setenv.sh (UNIX) file that are passed to Java at startup to allow the system to render graphics. PRPC requires this setting in order to display graphical reports.

Add the JVM systems properties setting **-Djava.awt.headless=true** to the JVM options for your Tomcat server in the same way you added the memory settings above. See “Setting JVM Memory Parameters” on page 17.

For example, if you created a setenv file to set the memory parameters, add or modify the JAVA_OPTS setting in the setenv.bat (Windows) or the setenv.sh (UNIX) file to include:

```
–Djava.awt.headless=true
```

For example, your settings may look similar to the following:

```
# Change -Xmx value based on available physical memory
JAVA_OPTS=”-Xms768m –Xmx1024m –XX:PermSize=64m
–XX:MaxPermSize=160m -Djava.awt.headless=true”
```

For more information, see “Creating the setenv File” on page 18.

## Setting User ID and Password in Tomcat

If you installed Tomcat on UNIX, or on Windows using the .zip distribution, you must enter user ID and password information into the **tomcat-users.xml** file in the ***TomcatInstallDirectory*/conf** directory. Windows users who use the Tomcat setup.exe file to install should also verify the user name and password created in this file.

This file sets some default roles for users logging onto the Tomcat server:

```
<?xml version='1.0' encoding='utf-8'?>
<tomcat-users>
  <role rolename="tomcat"/>
  <role rolename="role1"/>
  <user username="tomcat" password="tomcat" roles="tomcat"/>
  <user username="role1" password="tomcat" roles="role1"/>
  <user username="both" password="tomcat" roles="tomcat,role1"/>
</tomcat-users>
```

Add an **administrator** and **manager** role, as follows:

```
<?xml version='1.0' encoding='utf-8'?>
<tomcat-users>
  <role rolename="tomcat"/>
  <role rolename="role1"/>
  <role rolename="manager"/>
  <role rolename="admin"/>
  <user username="tomcat" password="tomcat" roles="tomcat"/>
  <user username="role1" password="tomcat" roles="role1"/>
  <user username="both" password="tomcat" roles="tomcat,role1"/>
  <user username="admin" password="admin" roles="admin,manager"/>
</tomcat-users>
```

For more information on user configuration in Tomcat, see the Tomcat's online documentation.

## Enabling DNS Lookup

The Tracer in Process Commander requires that DNS lookup be enabled in order to display the name of the client host, rather than its IP address. To enable DNS lookup, set enableLookups="true". This setting is found in the server.xml file, located in the conf directory. An example is shown below.

```
<Connector enableLookups="true" />
```

## Setting Localization Libraries (MS SQL)

If you are using a MS SQL Server database with an extended character set, for example to support multiple languages, you must add the path to the file **charsets.jar** in the **lib** directory of the JVM that you are using for Process Commander to the **setclasspath** file in Tomcat. See the examples below.

The **charsets.jar** file is an optional feature of the JRE, though in most JVM installations it is included by default. If your JVM **lib** directory does not include **charsets.jar**, rerun the JVM installer, choose "custom installation," and select the "Support for additional locales" feature. On Windows you can access the installer by clicking Start > Settings > Control Panel > Add or Remove Programs and select the JRE update, for example "Java 6 Update 6."

Note: Though the filename, **charsets.jar**, is the same for all JVM versions, the file is different in each version of the JVM. Be sure to add the path to the **lib** directory in your current JVM environment, for example, C:\jdk1.6.0_06\jre\lib\charsets.jar, or to JAVA_HOME as in the examples below. If you have more than one JVM installed on the Process Commander system, be sure to set the classpath to the JRE being used by Process Commander.

### Windows server example – setclasspath.bat:

Original file:

```
rem Set standard CLASSPATH
rem Note that there are no quotes as we do not want to introduce random
rem quotes into the CLASSPATH
set CLASSPATH=%JAVA_HOME%\lib\tools.jar
```

With the addition:

```
rem Set standard CLASSPATH
rem Note that there are no quotes as we do not want to introduce random
rem quotes into the CLASSPATH
set
CLASSPATH=%JAVA_HOME%\lib\tools.jar;%JAVA_HOME%\jre\lib\charsets.jar
```

### UNIX server example – setclasspath.sh:

Original file:

```
Set standard CLASSPATH
CLASSPATH="$JAVA_HOME"/lib/tools.jar
#
```

With addition:

```
Set standard CLASSPATH
CLASSPATH="$JAVA_HOME"/lib/tools.jar:"$JAVA_HOME"/jre/lib/charsets.jar
#
```

## Configure Data Source Connection and ExplicitTempDir

Add two elements to the **Context** element in the **Context.xml** file in your Tomcat **conf** directory:

- A **Resource** element specifying the Process Commander database connection (jdbc/PegaRULES) and
- An **Environment** element specifying the location of the Process Commander temporary directory (url/initialization/explicittempdir).

Details on completing these elements are given below.

Note: If this Tomcat server will be dedicated to a single Process Commander instance, add these elements to the <Context> element in the **Context.xml** file in your Tomcat **conf** directory.

However, settings in this file are loaded by all web applications started on this Tomcat server. If you are deploying other Process Commander instances or other applications on this server, you can instead specify the <Context> element at the host or application level. For more information, see the Apache Tomcat Configuration Reference at http://tomcat.apache.org/tomcat-6.0-doc/config/index.html.

## Completing the Tomcat Datasource Elements for Process Commander

The following are example Resource and Environment elements for MS SQL. Values you should change for your environment and database type are highlighted. Click the highlighted link for details.

```
<Context>

<Resource name="jdbc/PegaRULES"
  auth="Container"
  type="javax.sql.DataSource"
  driverClassName="com.microsoft.sqlserver.jdbc.SQLServerDriver"
  url="jdbc:sqlserver://localhost:1182;databaseName=rules3;
       SelectMethod=cursor;SendStringParametersAsUnicode=false"
  username="pega3"
  password="pega"
  maxActive="100"
  maxIdle="30"
  maxWait="10000"
/>

<Environment name="url/initialization/explicittempdir"
value="D:/PRTemp6-1/prweb" type="java.lang.String" />

</Context>
```

In addition, you may need to add a **connectionProperties** attribute for Oracle or UDB databases. See below, for details.

### Specifying the *jdbc/PegaRULES* Resource element

Enter the connection information for your Process Commander database in the Resource element attributes as follows:

- **name**: enter “jdbc/PegaRULES”
- **auth**: enter “Container”
- **type**: enter “"javax.sql.DataSource"
- **driverClassName**: The fully qualified Java class name of the appropriate JDBC driver:
  - Oracle 9i/10g: **oracle.jdbc.OracleDriver**
  - IBM DB/2 Type 4: **com.ibm.db2.jcc.DB2Driver**
  - SQL Server 2005 and 2008: **com.microsoft.sqlserver.jdbc.SQLServerDriver**
- **url**: The connection URL for the Process Commander database in the following format for your database type:

- Oracle 9i/10g (Native OCI Client): **`jdbc:oracle:oci:@your_tns_name`**
- Oracle 9i/10g (Thin Client): jdbc:oracle:thin:@myServer:1521:myDatabase
- DB/2 (Universal Driver): jdbc:db2://serverName:port/dbName
- SQL Server 2005 and 2008: jdbc:sqlserver:// <*your_sql_server_host*>:1433;databaseName=<*your-database_name*>;SelectMethod=cursor;SendStringParametersAsUnicode=false

> Note: Be sure to include the **SelectMethod** and **SendStringParametersAsUnicode** parameters in URL for MS SQL 2005 or 2008.

### Note: Using MS SQL 2008

As described below, configure MS SQL 2008 with the same driver and connection URL as 2005. However, you should be aware of two differences:

1. You must enable the TCP/IP protocol for SQL Server:
   - Start the MS SQL Server Configuration Manager
   - Expand **SQL Server Network Configuration**, and click **Protocols for SQLEXPRESS**.
   - In the list of protocols, right-click **TCP/IP** and select **Enable**.

   Leave the window open.
2. The typical default TCP/IP port number used in SQL 2005, 1433, is changed in 2008 to a host-specific number. To find the port number for your machine:
   - In the list of protocols, double-click **TCP/IP** to open the TCP/IP Properties window.
   - Display the **IP Addresses** tab, and find the **TCP Dynamic Ports** setting. This is the port number to use in the database connection URL.

- **username** and **password**: Enter the user name and password you created for the PRPC user on your database.
- **maxActive**, **maxIdle**, and **maxWait**: These attributes control the processing of connection in the Tomcat database connection pool. Enter the following values as defaults:
  - maxActive="100"
  - maxIdle="30"
  - maxWait="10000"

### Add any connection properties required by your database

Add a **connectionProperties** attribute to the Resource element to specify any connections properties required by your database:

- **Oracle:**

  If you are using Java 5 and the ojdbc5.jar driver file, add the connection property o**racle.jdbc.V8Compatible=true**. For example:

```
connectionProperties="oracle.jdbc.V8Compatible=true"
```

  **Note:** If you are using Java 6 and the ojdbc6.jar driver file, the V8Compatible setting is not required.

- **UDB:**

  For UDB, you must specify the current Schema by listing the **currentSchema**, property specifying the DB user schema name, and the **currentFunctionPath** property specifying SYSIBM,SYSFUN,and the DB user schema name.

```
connectionProperties=currentSchema=<SCHEMANAME>,
currentFunctionPath=SYSIBM,SYSFUN,<SCHEMANAME>
```

  For example, if the schema name is PRPC, the connectionProperties attribute would be:

```
connectionProperties="currentSchema=PRPC;
currentFunctionPath=SYSIBM,SYSFUN,PRPC"
```

  Note: Be sure to enter the schema name in all capital letters, as in the example.

  In addition, DB2 UDB versions 9.5 and 9.7 require four settings:

  – fullyMaterializeLobData=true
  – fullyMaterializeInputStreams=true
  – progressiveStreaming=2
  – progressiveLocators=2

  You can add these settings as additional connection properties or as parameters in the URL.

  To specify these settings as connectionProperties, add these to the connection properties attribute, separating each setting with a semi-colon:

```
connectionProperties=currentSchema=<SCHEMANAME>,
currentFunctionPath=SYSIBM,SYSFUN,<SCHEMANAME>;
fullyMaterializeLobData=true;
fullyMaterializeInputStreams=true;
progressiveStreaming=2;progressiveLocators=2
```

To add these settings as parameters to the conncection URL, append them to the URL as semi-colon separated properties:

```
url=jdbc:db2://sysvol2:50000/PRPC:fullyMaterializeLobData=true;
fullyMaterializeInputStreams=true;
progressiveStreaming=2;progressiveLocators=2;
```

> Note: The currentSchema property must be specified as a connectionProperties setting, and cannot be entered into the URL.

### Specifying the Process Commander temporary directory (url/initialization/explicittempdir)

Specify the location for a Process Commander temporary directory in Environment element attributes as follows:

<Environment name="url/initialization/explicittempdir"
value="<*your_Process_Commander_tempdir*>" type="java.lang.String" />

- **name**: enter “url/initialization/explicittempdir”
- **value**: enter the full path to a directory dedicated to Process Commander temp space.

> Note: This setting is case-sensitive. Be sure to use the exact directory names in the path.

**type**: enter “"java.lang.String "

> Note: Process Commander uses this directory as a staging directory for archive imports, as storage for Index and other static content, for on-disk Rules caches, and for temporary storage. It is important to define a temp directory for Process Commander, separate from your system or application server temporary directory, with full read/write access.

## Stop and Restart the Server

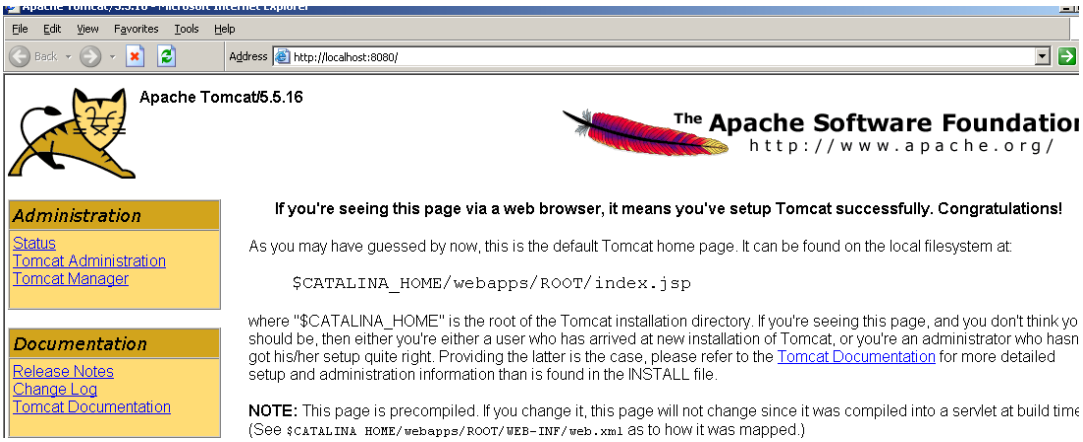
Stop and restart the Tomcat server to make these changes effective.

# Deploying the Process Commander Application WAR Files

## Start Tomcat and Open the Tomcat Home page

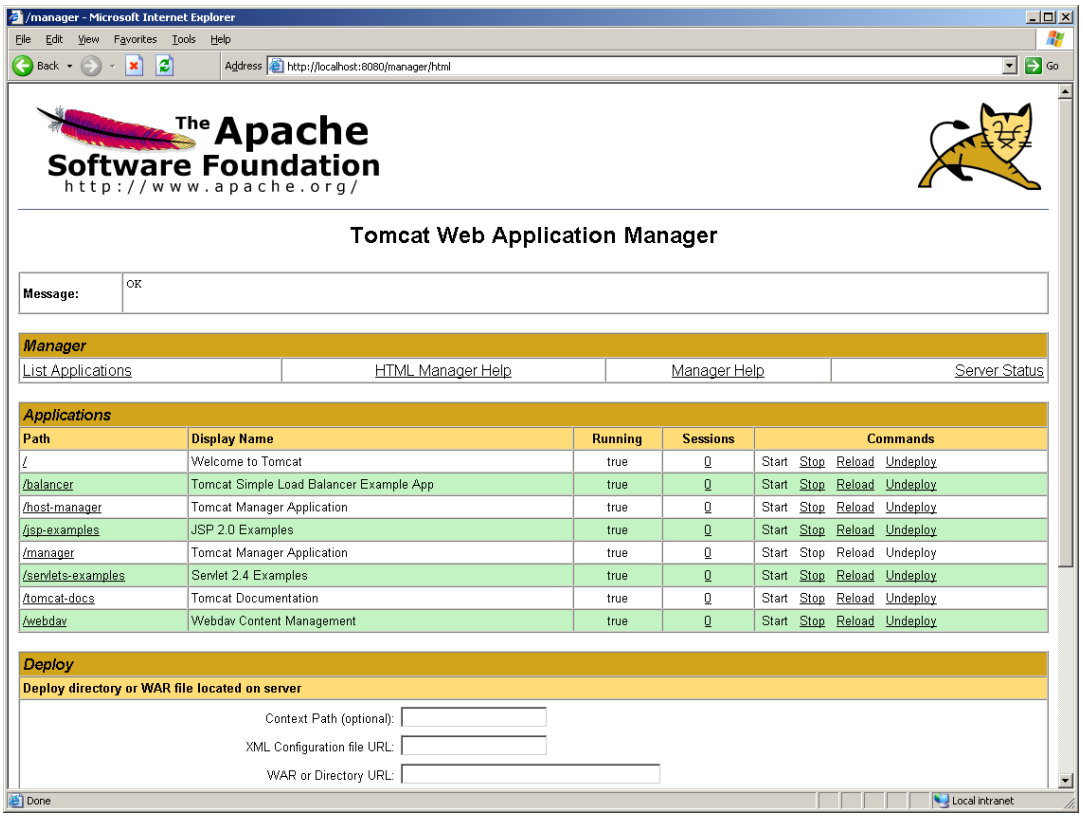
Start Tomcat and open the Tomcat home page in a web browser by entering the following URL:

http://localhost:8080

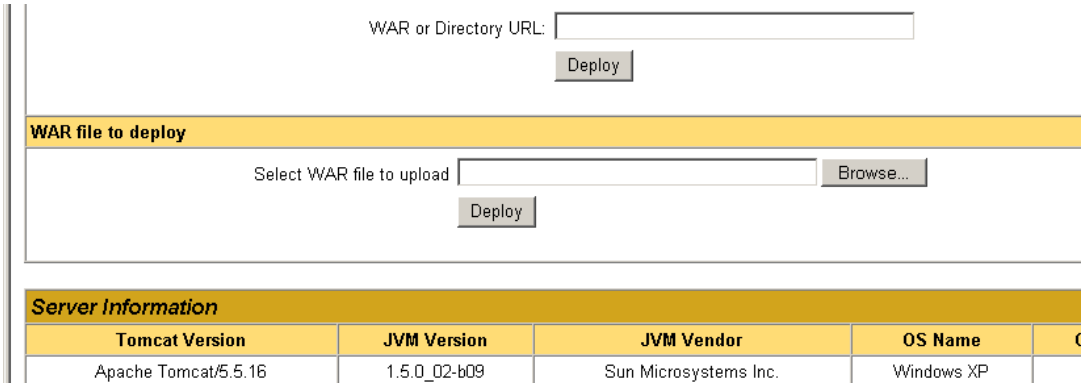



## Install the PegaRULES Process Commander WAR (prweb.war)

1. From the Tomcat home page, click on the **Tomcat Manager** link. At the login page, enter the user credentials you defined earlier in the tomcat-users.xml.




2. Scroll down to the Deploy section and locate the WAR file to deploy section.




3. In the section WAR file to deploy, click the Browse button to navigate to the location of the prweb.war file and select it.
4. When the prweb.war file is displayed in the **Select WAR file to upload field**, click the **Deploy** button. Tomcat creates the context root **\<*tomcat_install_root*>\webapps\prweb** and expands the WAR file there.

The application also appears in the list of deployed applications in the Manager.


5. Continue with the deployment of the prsysmgmt.war and prhelp.war.

## Deploying the System Management and Online Help Applications

Deploy the online help application (prhelp.war) and System Management application (prsysmgmt.war) in the same manner as prweb.war.

For more details on the System Management Application, refer to the *System Management Application Reference Guide*.

# Assigning the PegaDiagnosticUser Role to your System Management Application Users

Users accessing the System Management Application in Process Commander must be assigned the role PegaDiagnosticUser. If the user does not have this role they will be challenged for authentication when they attempt to access the System Management Application from Process Commander.

The role PegaDiagnosticUser is associated with the System Management Application through a security-constraint element on the Diagnostic Data servlet defined in the prweb.war application's web.xml file:

```
- <security-constraint>
  - <web-resource-collection>
      <web-resource-name>Diagnostic Data</web-resource-name>
      <description>Serves diagnostic files generated by the JMX
        client</description>
      <url-pattern>/DiagnosticData</url-pattern>
      <http-method>GET</http-method>
      <http-method>POST</http-method>
    </web-resource-collection>
- <auth-constraint>
    <role-name>PegaDiagnosticUser</role-name>
  </auth-constraint>
</security-constraint>
```

If you do not want to restrict access to the System Management Application, edit the web.xml file in the WEB-INF directory of the prweb.war deployment in your application server and delete this security constraint element.

> Note: Changing this setting in the web.xml file may require redeployment of the PegaRULES Web Application.

If you want to restrict access to the System Management Application, use your application server tools to assign the PegaDiagnosticUser role to any users that must be able to access the System Management Application.

## Next Steps

If you have not completed the Rulebase Loading phase, continue with the next section: Steps 4-9 Rulebase Setup Phase. Be sure that the Process Commander applications are not started.

If you have completed the Rulebase Loading phase, start the Process Commander applications and continue with Step 10, Final Settings in PRPC.

# Steps 4 – 9:

# Rulebase Setup

## Install the Process Commander Schema and Rules in the Database

Run the Process Commander Installation and Upgrade Assistant to create the Process Commander schema in the database, and load the Process Commander application rules.

> Note: The following procedure describes the use of the user-interface to the installer to create the schema in the database and load the base Process Commander rules. The 6.1 SP2 distribution also includes scripts to support these tasks through a command-line interface. Support is provided through install.bat and install.ksh files configured through a properties file, setupDatabase.properties. To use the scripted option, see "Chapter 6: Command-line Interfaces for Installation, Upgrade, and Utilities."

1. Download the Process Commander image, **PRPC_DistributionImage.zip**, or copy it from your CD, to the computer on which you will run the installation assistant.

   The computer on which you run the assistant must have access to allow a database connection to the Process Commander database. This computer should also have 1GB of RAM and at least 2.5 GB of disk space available to the system temporary directory.

---

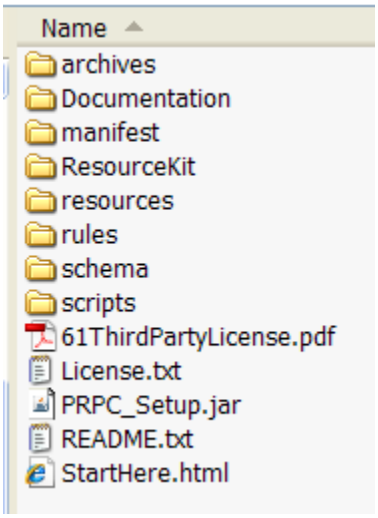
Note: On a 64-bit system, running PRPC_Setup.jar requires up to 1.6 GB memory. To avoid memory errors, use one of the following methods:

1. Use a 32-bit JVM to run PRPC_Setup.jar, or
2. Increase the memory allocated to the JVM:
   - In the **scripts** directory of your Process Commander software distribution, open the file setupDatabase.xml in a text editor.
   - In the file find the following element:

     ```
     <!-- Misc. Properties to tweak processing profile -->
     <property name="jvm.arg.mem.max" value="-Xmx1024m"/>
     ```

   - Increase the value of –Xmx to 2048.
   - Save and close the file

2. Extract **PRPC_DistributionImage.zip** to a directory, to create the following directory structure:




3. Double click on the **PRPC_Setup.jar** file or right-click the file and click **Open With > Java**, to launch the Installer. If JAR files are not associated with Java commands on your system, launch the installer at the command-line.

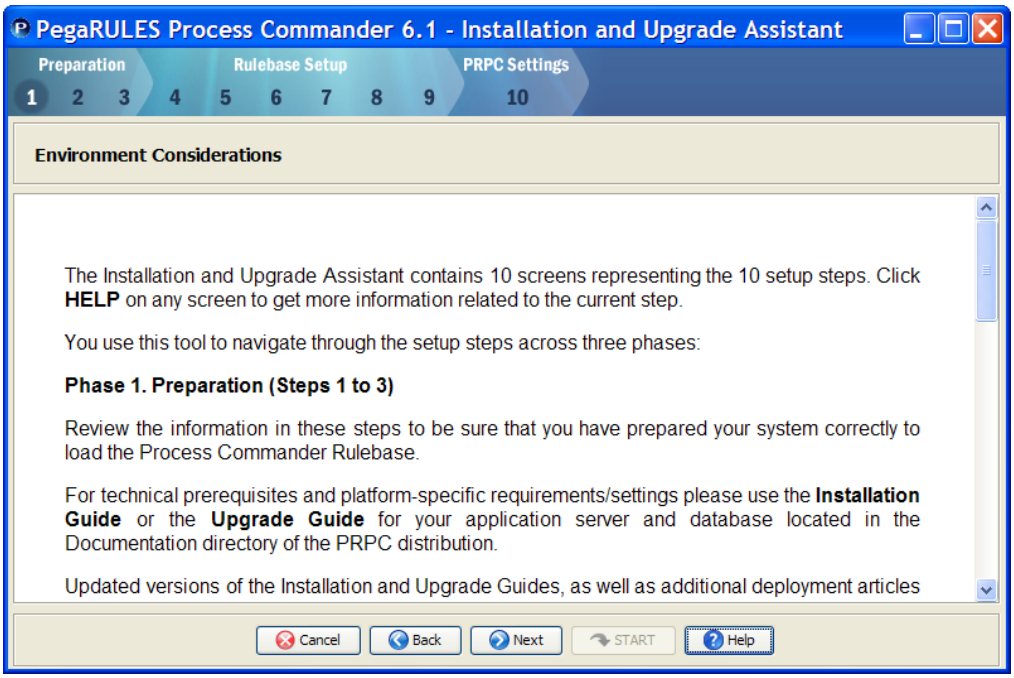
   In a command-line console, change to the directory containing PRPC_Setup.jar, and type in the following command:

   ```
   Java –jar PRPC_Setup.jar
   ```

Note: On Windows systems you may see the following error when you try to launch PRPC_Setup by double-clicking the JAR file.

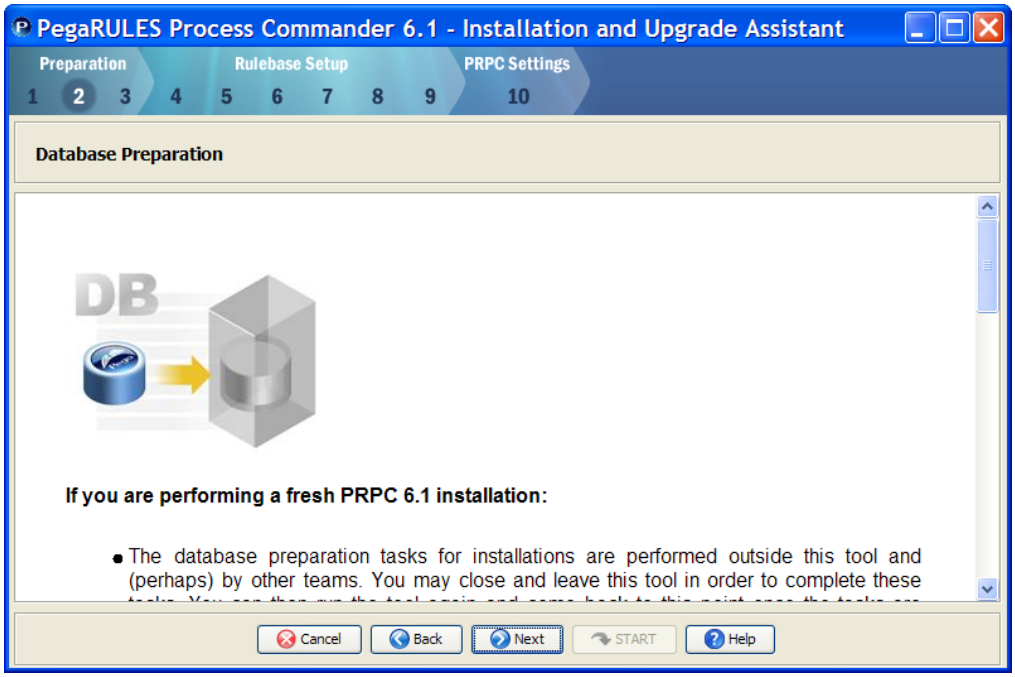
```
Could not extract JAR file to directory:
C:\Windows\System32\pegaGUIInstallerTemp
```


4. Click **Next** to display the Environment Considerations.




See “Step 1, Environment Considerations in this guide for more information on completing this step.

5. Click **Next** to display the Database Preparation page.

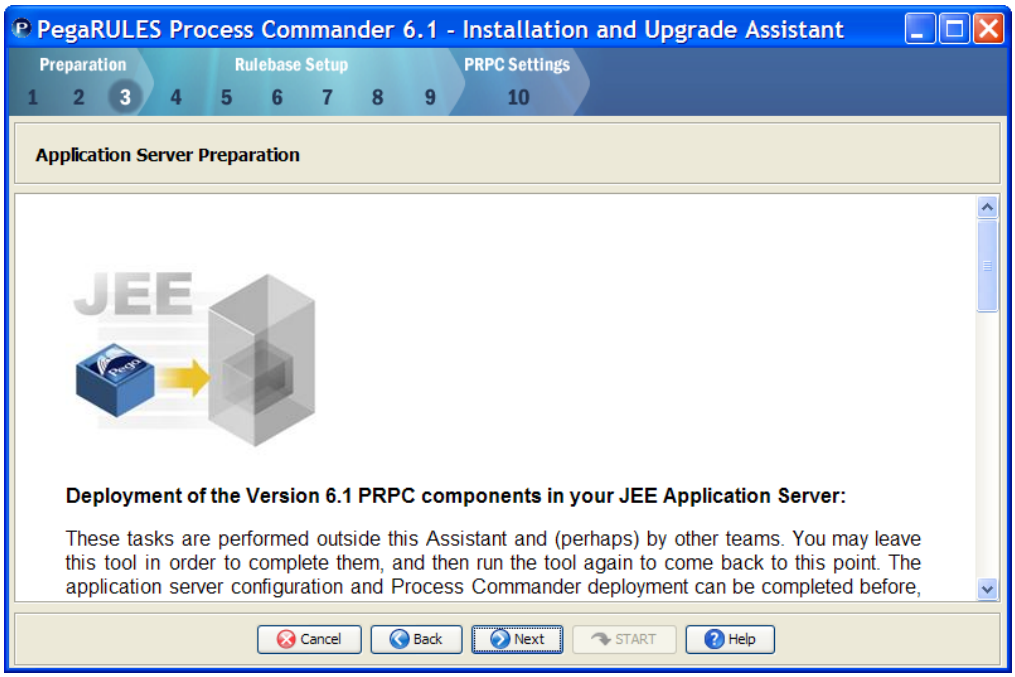



6. Review this page to confirm that you have prepared the database as required.

   See “Step 2, Database Preparation” in this guide for more information on completing this step.

   These tasks are completed outside of the Assistant. You must complete this preparation before running the Rulebase Load phase. You can exit from this tool to complete them, and then restart to continue beyond this point.

7. When your Database Preparation is complete, click **Next** to display Application Server Preparation.

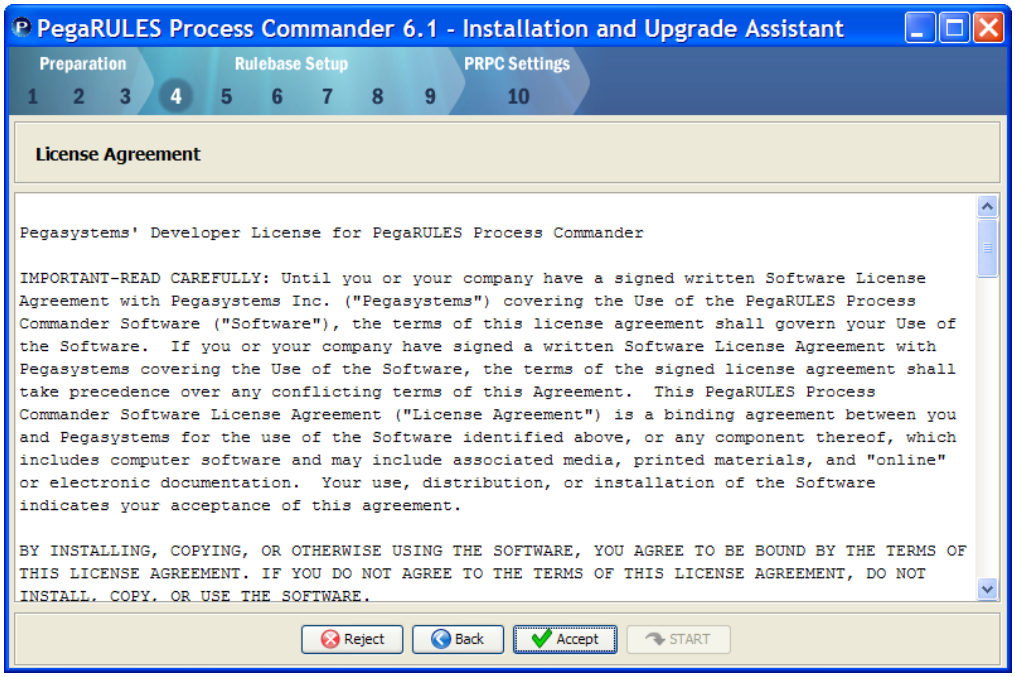



8. Review this page to learn about the tasks you complete to prepare your application server before deploying the Process Commander applications.

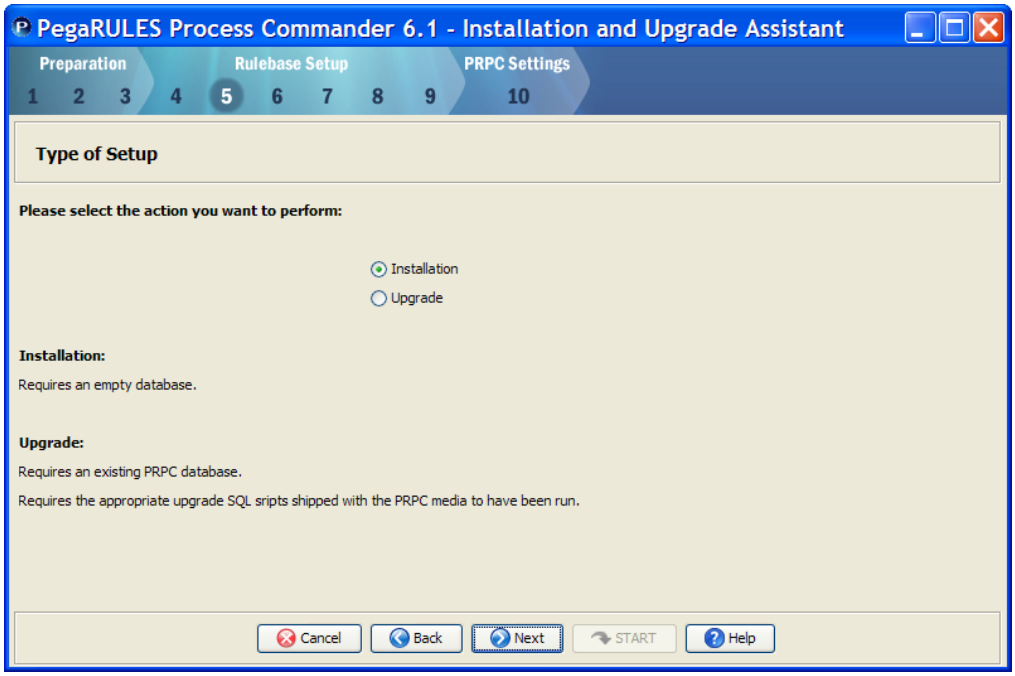
   These tasks are completed outside this Assistant. You can exit from this tool to complete them, and then restart to continue beyond this point, or you can continue with the Rulebase Setup phase immediately and complete the Application Server Preparation later.

> Note: When the Database Preparation is complete, Step 3, the Application Server Preparation, and Steps 4-9 in the Rulebase Setup Phase can be performed in either order or simultaneously. You can, for example, launch the Rulebase Setup process in the Assistant and then configure your Application Server while the Rulebase load process is running.
>
> However, you must configure the application server and deploy the Process Commander applications before you can complete Step 10, PRPC Final Settings.
>
> Important: You must be careful not to start the deployed Process Commander applications while the Rulebase loading process is still running on the database. Note that your application server may start the application automatically when it is deployed. If the Process Commander application attempts to connect to the database while the Rulebase is loading, the database loading will not complete correctly.

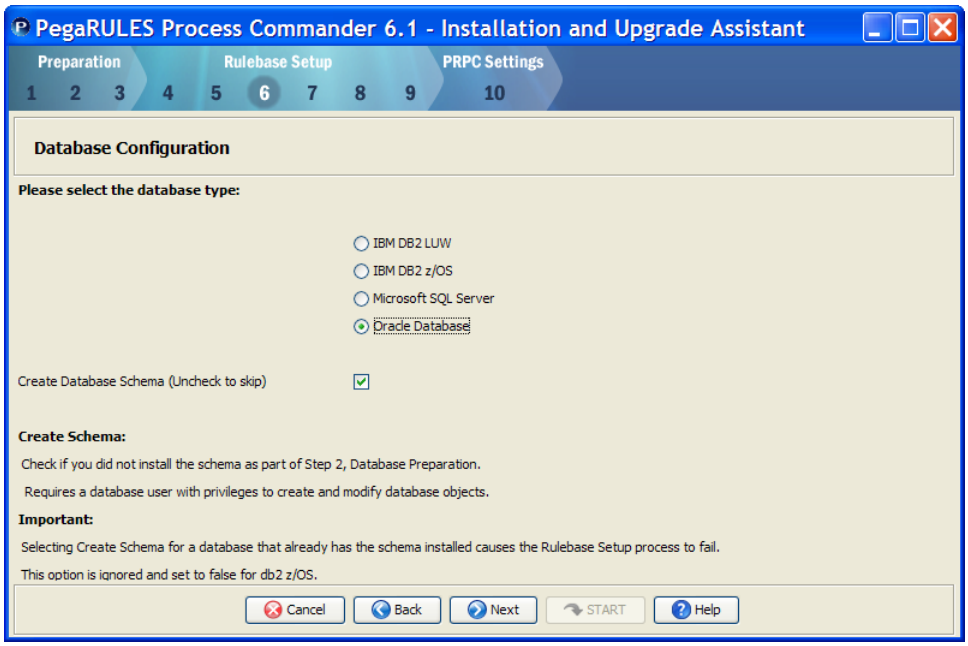
9. If you have completed your Application Server Preparation or you want to continue with the Rulebase Setup in the database before completing the Application Server Preparation, click **Next** to display the License Agreement.




10. Review the License agreement and click **Accept** to continue.



11. Select **Installation** and click **Next** to display the **Database Configuration** page.




12. Complete this page as follows:

**Database Type**

– Select the type of your Process Commander database.

> Note: Instructions to install using a database on DB2 on z/OS are not covered in this guide. See the *Installation Guide for DB2 on z/OS and WebSphere* for information on this platform.

**Create Database Schema**

– Leave **Create Database Schema** checked to allow the Assistant to create the appropriate schema in the database as part of the Rulebase Load process. You database should be empty.

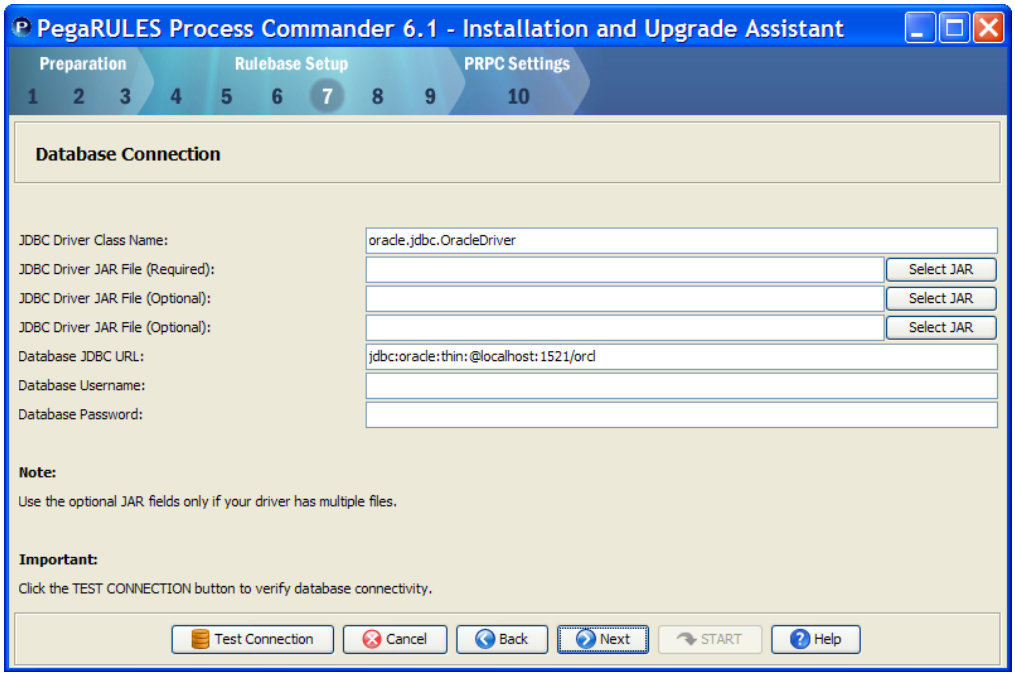
– Uncheck **Create Database Schema** if you used your database tools to create the schema using the SQL scripts provided in the **schema** directory of your software distribution.

See “Using a SQL Script to Create the Schema” in this guide for more information.

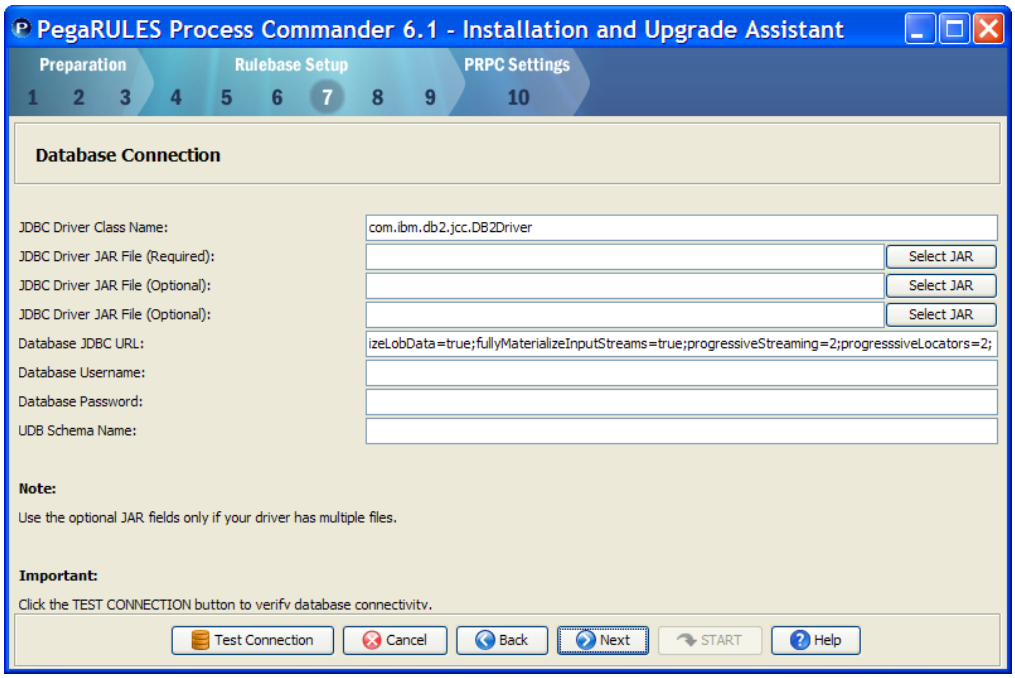
> Note: If you run the Assistant with Create Database Schema selected on a database that already has the schema installed, the Rulebase Load process fails with an error message.

Click **Next** to display the Database Connection page.

If your database type is Oracle or MS SQL, you specify the JDBC driver class name, the location of the driver JAR file, the database connection URL, and the database user name and password.




If your database type is IBM UDB, you also specify the user schema name:



13. Complete the fields to specify the database connection to your Process Commander database:
    - **JDBC Driver Class Name** – Enter the name of the appropriate database driver class:
        - Oracle 10g/11g: **oracle.jdbc.OracleDriver**
        - IBM DB/2 Type 4: **com.ibm.db2.jcc.DB2Driver**
        - SQL Server 2005 and 2008: **com.microsoft.sqlserver.jdbc.SQLServerDriver**
    - **Driver JAR file** – Enter the full path to the appropriate driver file for your database type and version:

      Use additional fields if your database requires more than one JAR file.

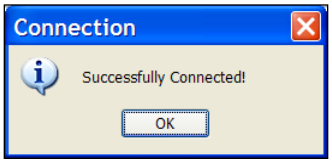
      See “Install Database Drivers on your Application Server” on page 15 for information on your database.
    - **Database JDBC URL** – Enter the JDBC connection URL for your Process Commander database. The formats for each database type are as follows:
        - Oracle 10g/11g (Native OCI Client): **`jdbc:oracle:oci:@your_tns_name`**
        - Oracle 10g/11g (Thin Client): **`jdbc:oracle:thin:@yourServer:1521:yourDatabase`**
        - DB/2 (Universal Driver): **`jdbc:db2://serverName:port/dbName`**
        - SQL Server 2005 and 2008:
        - **`jdbc:sqlserver://<your_sql_server_host>:1433;SendStringParametersAsUnicode=false;SelectMethod=cursor`**

> Note: The port number for MS SQL 2008 may be different than the typical default for 2005. See “Note: Using MS SQL 2008” on page 24 for more information.
>
> In the URL for MS SQL, be sure to include the SendStringParametersAsUnicode and Select Method parameters as in the example.
>
> In the URL for UDB, be sure to include the following parameters:
> - fullyMaterializeLobData=true;
> - fullyMaterializeInputStreams=true;
> - progressiveStreaming=2;
> - progresssiveLocators=2;

- **Username** and **Password** – Enter the user name and password you created for the Process Commander user on your database
- **UDB Schema Name**—For UDB databases, enter the schema name assigned to the Process Commander database when it was created.

14. When you have completed entering the database connection information, click **Test Connection**.

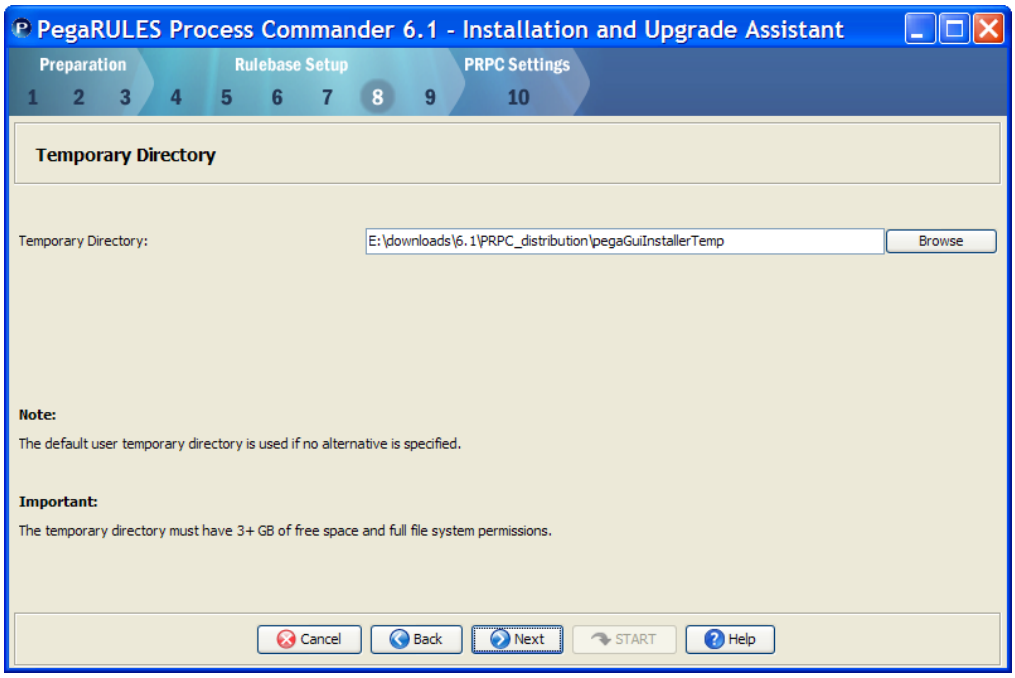
    The Installation Assistant tests whether it is able to make a connection to the database you have specified and displays the result:




    Click **OK** to close the window.

    If the connection is not successful, review your connection information, correct any errors, and retest.

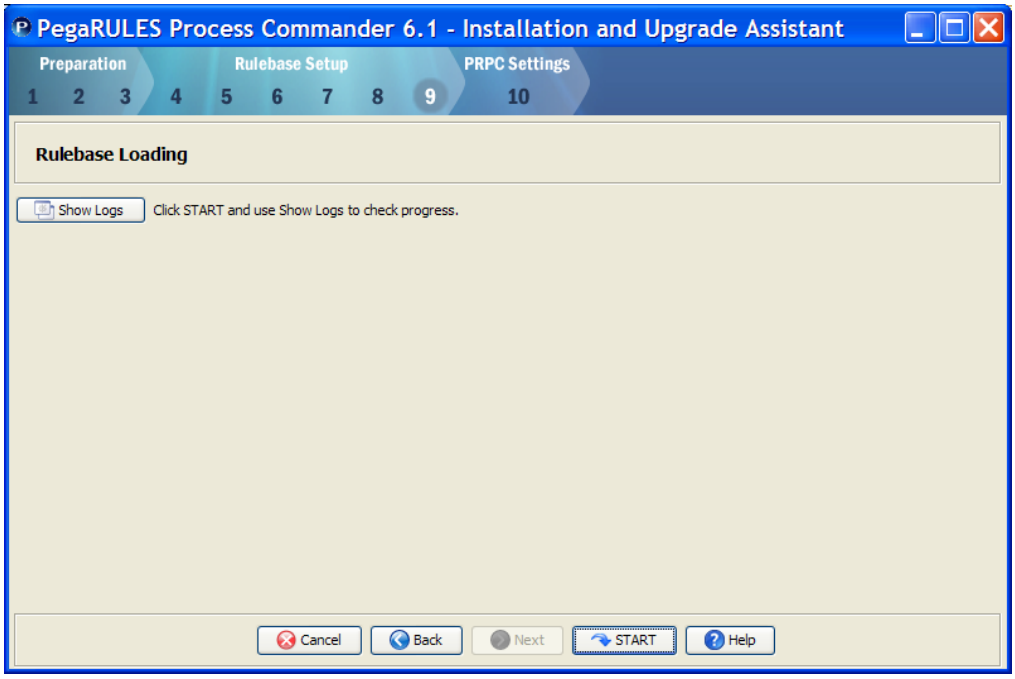
    If the connection is successful, click **Next** to continue.




15. Enter or **Browse** to select a temporary directory for the Installation Assistant to use while loading the Process Commander rules. By default, the Assistant uses a directory, **pegaGuiInstallerTemp**, created in the directory from which the tool is run. If you leave the field blank, the Installation Assistant uses the user system temporary directory.

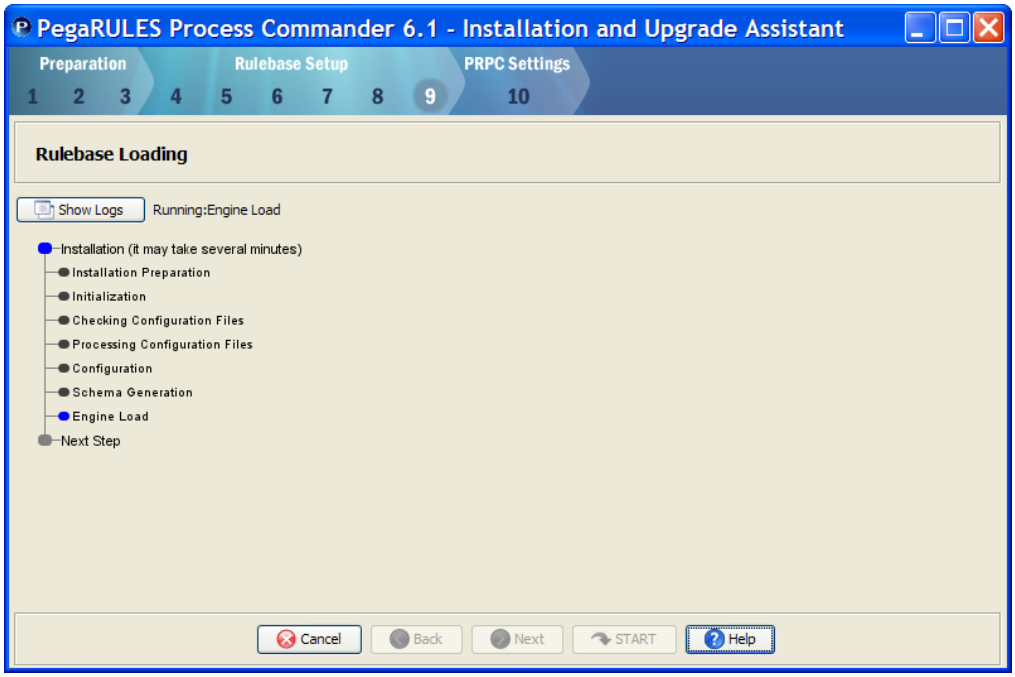
> **Important:** The Assistant requires at least 3 GB of space during the installation process. Be sure there is enough disk space to allow the directory to grow to this size. The Assistant deletes all temporary files at the end of the process.

– Click **Next** to display the Rulebase Loading page.

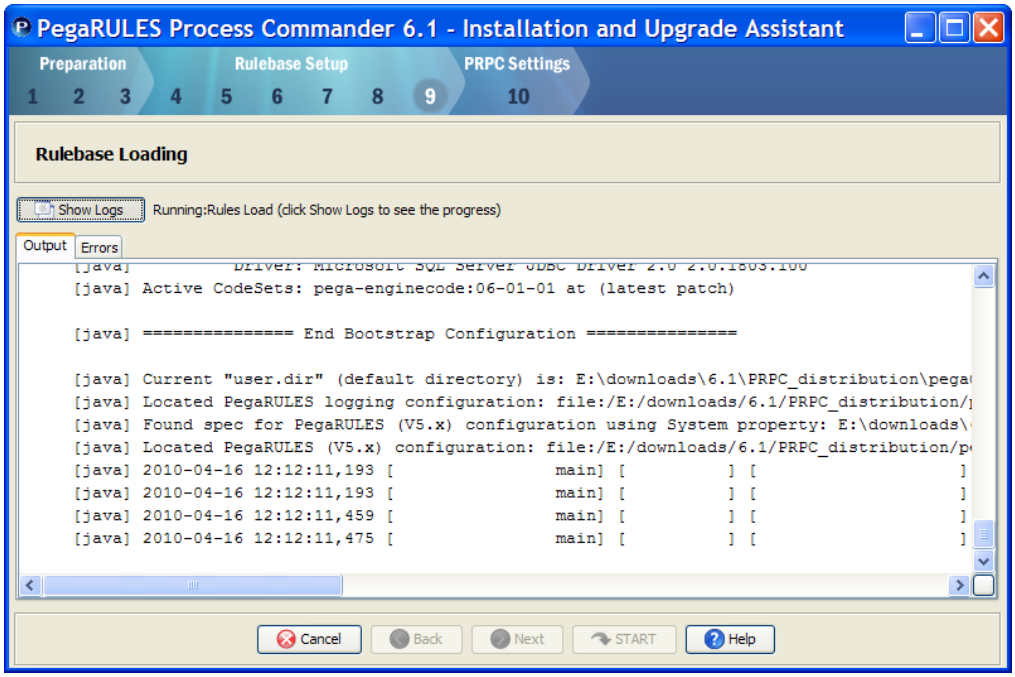



16. Press **START** to begin loading the database with the rules.

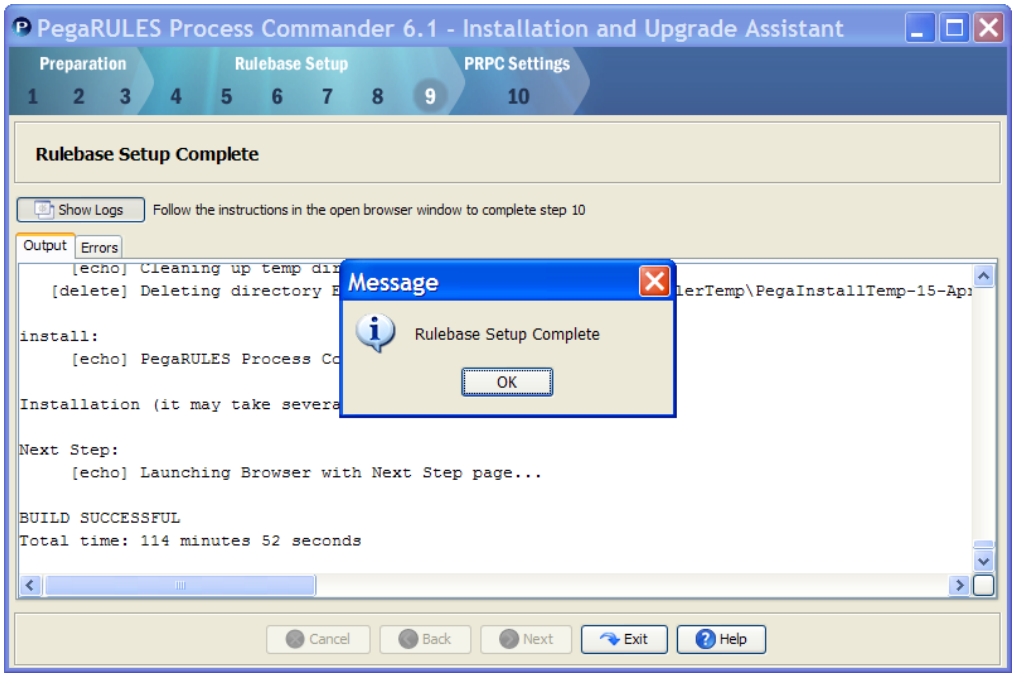
    The Assistant shows the progress through the stages of the installation:




Click **Show Logs** to see the log detail as the process is running.




The Assistant displays the following window when complete:




> Note: A typical installation can take 20 to 60 minutes depending on the speed of your system. If you choose to pre-load the assembled rules cache, the process takes an additional 90 minutes.
>
> When processing the larger files, the log can appear inactive for some minutes. Please allow the Assistant to finish.

**If the process does not complete:**

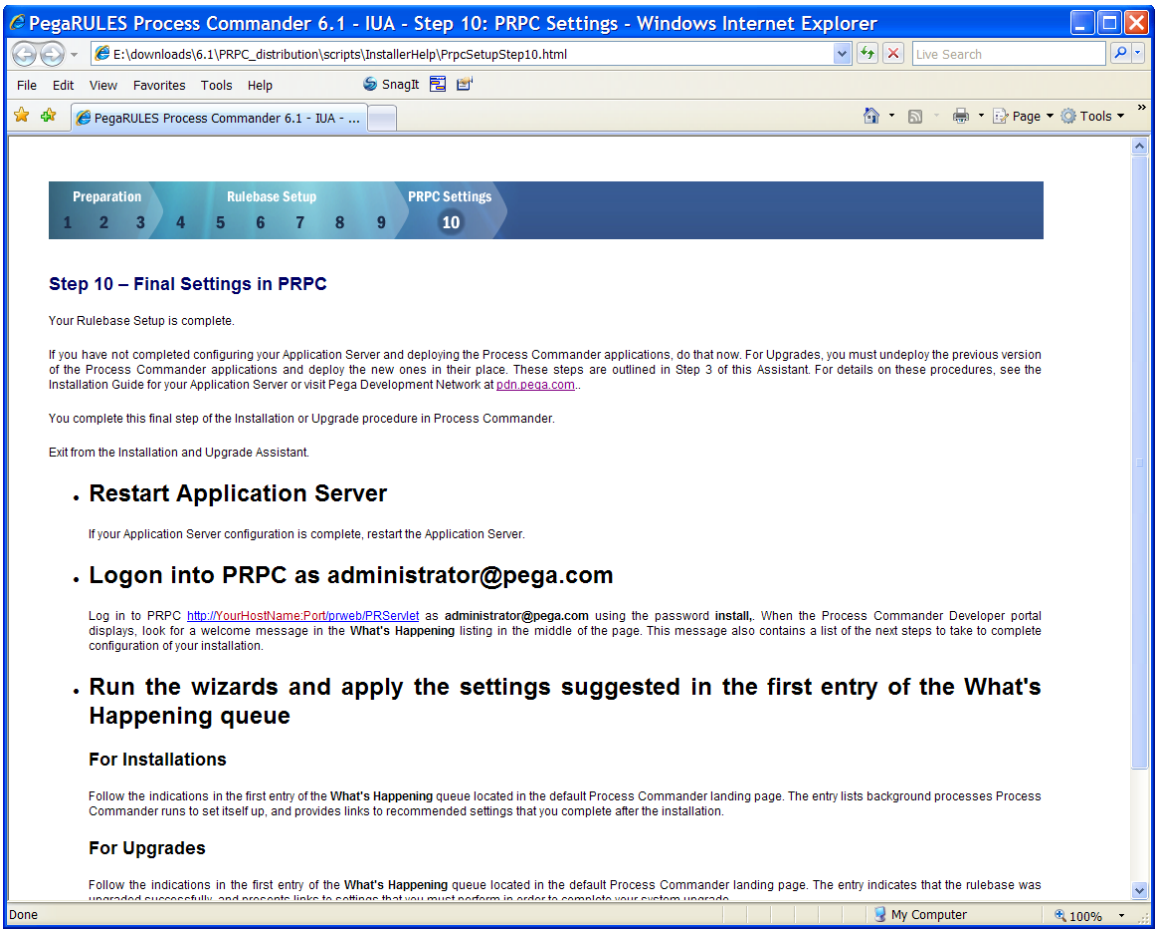
- Review information on the logs in the **Output** and **Error** tabs, and correct the problems.
- Exit from the Assistant.
- Clear the Process Commander database:

  Using your database tools, run the drop SQL script in the **schema\<*db-type*>\util** directory for your database type in the Process Commander software distribution, for example:

```
C:\PRPC_distribution\schema\oracledate\util\oracledate_drop_6.1.sql
```

- Restart this **Installation and Upgrade Assistant** and complete the steps again as directed above.

17. When the process completes, the instructions for **Step 10—Final Settings in PRPC** opens in a browser window:




18. Click **Exit** in the Assistant and continue with the procedures in the next section “Step 10—Final Settings in PRPC.

# Step 10:

# Final Settings in Process Commander

Access the Process Commander servlet, PRServlet, through a browser. Type in the URL for Process Commander application: http://<server>:<portnumber>/<context_root>/PRServlet. For example:

http://prpc-server:8080/prweb/PRServlet

The PegaRULES login screen (PRServlet) will be displayed:

Login to Process Commander:

- User Name: **administrator@pega.com**
- Password: **install**

The Process Commander Developer portal displays with a welcome message in the **What's Happening** section of thepage:

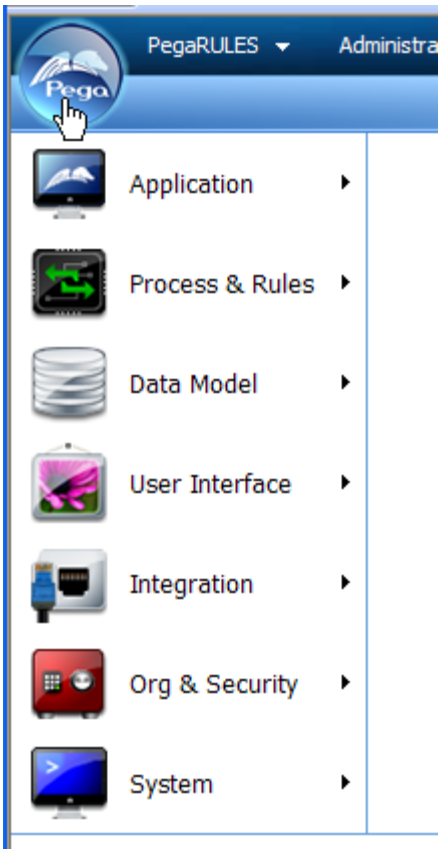

Process Commander automatically launches a process to index the Process Commander rules in the system to support full-text search in the Developer environment. While this process is running, you may notice some delay in the responsiveness of the Process Commander user interface. Depending on your system, the indexing will complete in 10 to 15 minutes.

> Note: This indexing process starts as soon as you start the Process Commander application. Before you login, you may notice this activity in your application server administrative console.

If you want to monitor the progress of the process, click here in the **What's Happening** message.

## Configure your environment

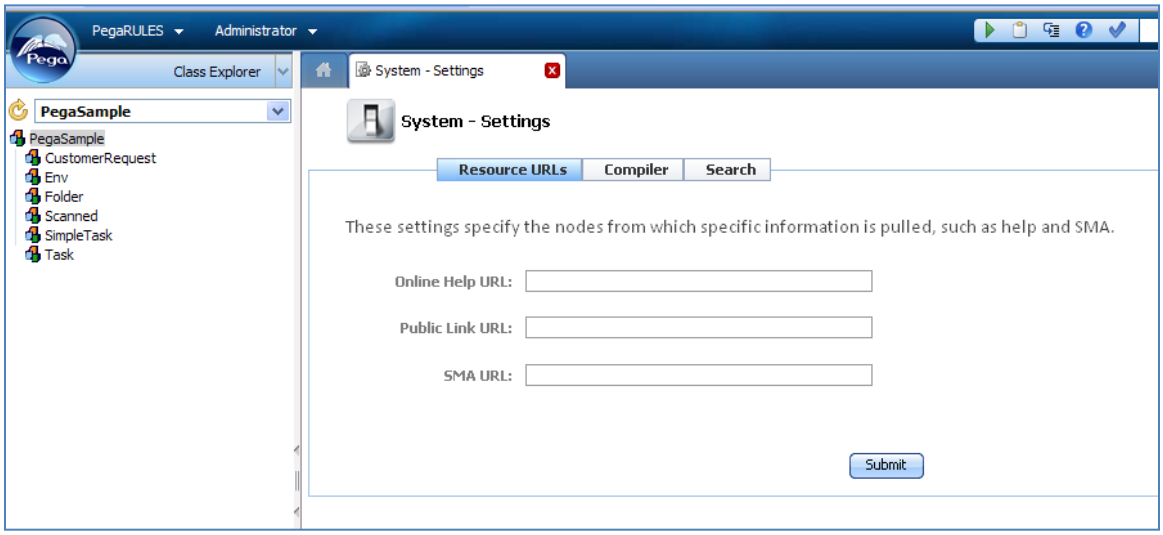
Use the Process Commander Landing pages, accessed from the Pega button, to configure your initial environment:




## Enable Developer Help and the System Management Application

To enable Help and the System Management Application, you must set the URLs that are used to access them.

1. On the Landing pages, select System -> Settings -> URLs



2. Complete this page to specify the URLs to access the Help (prhelp.war) and System Management (prsysmgmt.war) applications
   - In the field **Online Help URL**, enter the URL for the prhelp.war file:

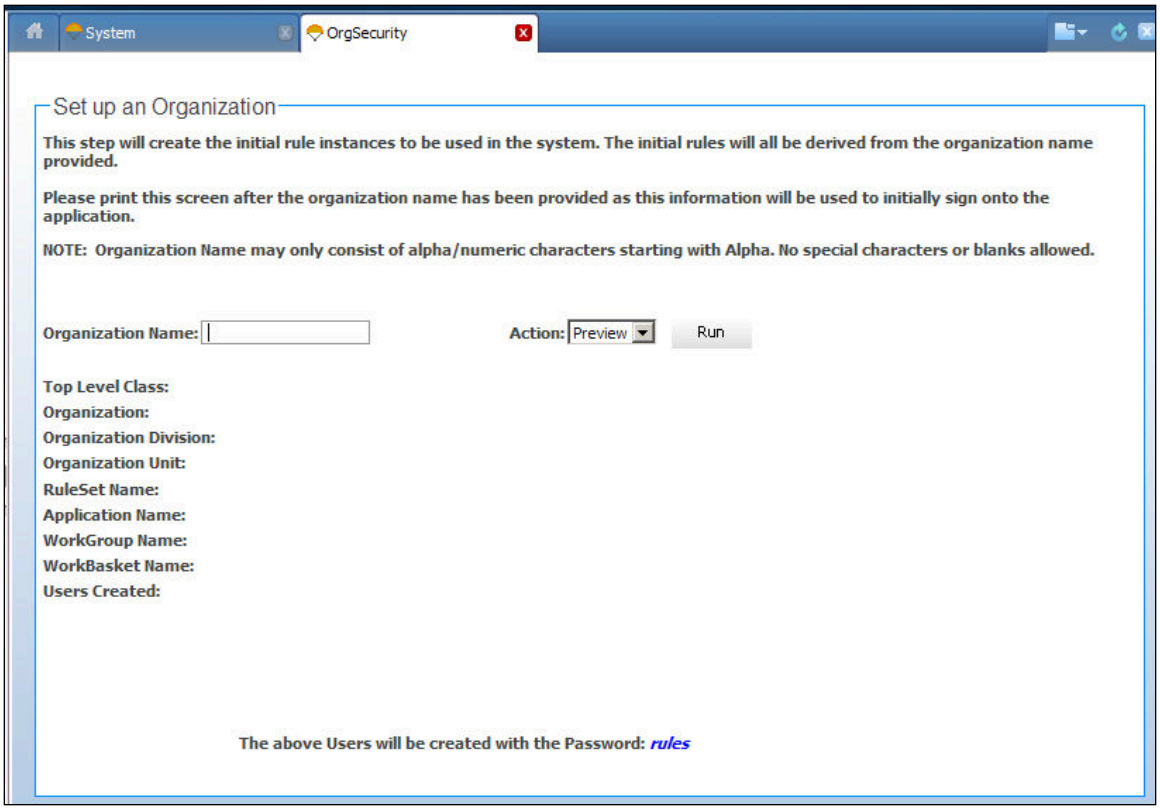
     Provide the name of your Process Commander server and the port number, for example, http://prpc_server:8080/prhelp
   - In the field for **SMA URL**, enter the URL for the System Management Application, prsysmgmt.war, for example, http://prpc_server:8080/prsysmgmt.
   - Click **Submit**
3. You must log out and log back in before launching the online help or the System Management Application.

   > Note: If you specify only *localhost* as the server name, only users logged into the same server on which the applications are deployed are able to access the applications. Replace *localhost* with the actual server name if you want to allow access to the applications from other Process Commander nodes.

   These applications are accessed from the Process Commander interface. Either application can be deployed to a different application server than the core Process Commander application.

## Create an Organization

1. Open the Org & Security > Organization Setup landing page:




2. In the **Organization Name** field, enter the name of your organization, leave the **Action** control set to **Preview** and click **Run** to review the objects and users that will be created:

   – Right-click in the window and select Print to print a record of this page.

> Important: Be sure to preview and record the information on this page. The organization name you provide will be used to create the initial users and access groups for your Process Commander system. You will need to know the user names to log on and begin using your Process Commander installation.

3. Set the **Action** to **Create** and click **Run** to create the organizational structure.

4. Stop and restart the application server.

Continue with the next section to enable the System Management Application for your installation.

# Install Solution Frameworks

If you are installing one of Pegasystems Solution Frameworks as part of your Process Commander development environment, install the framework now. Follow the instructions in the Installation Guide for the framework.

# *Appendix A:* Command-line Interfaces for Installation and Upgrade

Process Commander 6.1 SP2 provides a command-line interface for installing or upgrading Process Commander databases. You can call t mander Schema and Database

This section describes the command-line interface to the Process Commander Installation and Upgrade Assistant.

To install the Process Commander database, you

1. Create a database using your DB vendor's tools
2. Specify the database connection information in the **setupDatabase.properties** file
3. Run the script **install.bat** or **install.ksh** to create the Process Commander schema in the database and load the initial Process Commander rules and classes.

To complete the Process Commander installation, you must also configure your application server and deploy the Process Commander applications. These scripts do not perform any application server operations.

For complete information on installing Process Commander, see the Process Commander Installation Guide v6.1 SP2 for your database and application server. This guide is available in the Doc directory in your software distribution and on the Pega Developer Network, pdn.pega.com.

To upgrade a Process Commander database, you

1. Use your database tools to run SQL scripts that Pegasystems provides to update the database schema
2. Specify the database connection information in the **setupDatabase.properties** file.
3. Run **upgrade.bat** or **upgrade.ksh** to update the Process Commander rules and classes

> Note: Executing upgrade.bat or upgrade.ksh does not update the database to the 6.1 schema. You must apply the SQL scripts provided by Pegasystems to your Process Commander database before running the upgrade script.

For complete information on upgrading Process Commander, see the PegaRULES Process Commander Release V6.1 SP2 Upgrade Guide 5.x to 6.1 SP2. This guide is available in the Doc directory of your software distribution and on the Pega Developer Network, pdn.pega.com.

## Prerequisites

Before running these scripts, you must complete the following prerequisites:

1. You must have a JDK installed (Java 5 or later), and the path to that JDK must be defined in a **JAVA_HOME** environment variable.
2. You must have the JDBC driver JAR file for your target database available.
3. For an installation, you must create a database with a minimum of 2GB for the user tablespace and allow the database to grow. Create a user for this database with full read and write permissions to create tables, procedures, and views.

You provide connection information for this database to the script, including the location of the driver file, the driver class name, the database connection URL and the user's username and password.

## Installation and Upgrade Setup

Complete these steps before either installing or upgrading Process Commander.

1. Extract the contents of the coreBuildDistributionImage.zip into a directory.
2. Open the command-line console for your platform and change directory into the “scripts” subdirectory.
3. Edit the file setupDatabase.properties to provide the following property values:

– **pega.jdbc.driver.jar=**

The full path to the JDBC driver file for your database

The driver required for each database type is:

**Oracle 10g/11g:**

- For Java 5: ojdbc5.jar
- For Java 6: ojdbc6.jar

**DB/2 UDB:** db2jcc.jar

**SQL Server 2005,2008:**

- For Java 5: sqljdbc.jar
- For Java 6: sqljdbc4.jar.

– **pega.jdbc.driver.class=**

The class name of the JDBC driver for your database.

The class required for each database type is:

**Oracle 10g/11g:** oracle.jdbc.OracleDriver

**IBM DB/2 Type 4:** com.ibm.db2.jcc.DB2Driver

**SQL Server 2005, 2008:** com.microsoft.sqlserver.jdbc.SQLServerDriver

– **pega.database.type**=

The type of database:

**Oracle 10g/11g:** oracledate

**DB/2 UDB:** udb

**SQL Server 2005, 2008:** mssql

– **pega.jdbc.url=**

The connection URL for your database, for example:

**Oracle 10g/11g (Native OCI Client):** jdbc:oracle:oci:@your_tns_name

**Oracle 10g/11g (Thin Client):**
jdbc:oracle:thin:@myServer:1521:myDatabase

**DB/2 UDB:** jdbc:db2://serverName:port/dbName

**SQL Server 2005, 2008:**

```
jdbc:sqlserver:// <your_sql_server_host>:1433; databaseName=<your-
DBname>;SelectMethod=cursor; SendStringParametersAsUnicode=false;
```

> Note: Be sure to include the SelectMethod and SendStringParametersAsUnicode parameters for SQL Server.

– **pega.jdbc.username=**
**pega.jdbc.password=**

The user name and password to use when connecting to your database.

– **user.temp.dir=**

The full path to a temporary directory for the script process to use while loading the Process Commander rules. If not specified, the installer uses the user system temporary directory.

> Important: At least 3 GB of space is required during the installation process.

– **run.assembler=true/false**

If true, the installer pre-assembles the Process Commander Classes and makes them available in a cache.

If false, classes are assembled as they are requested during Process Commander use, slowing response time for users.

Pre-loading the cache makes the system perform better for the initial Process Commander users but adds up to 90 minutes to the installation process.

– **#bypass.pega.schema= (Install only)**

To skip schema creation in the Installation Assistant, remove the hash mark, #, at the beginning of this line in the setupDatabase.properties file. Use this option if you have previously created the schema in the database using the SQL scripts provided by Pegasystems.

Setting this property corresponds to clearing the Create Schema check-box in the Install Assistant user interface.

Important: The bypass.pega.schema property is ignored by the upgrade script. Before upgrading the database, you must apply SQL scripts, provided by Pegasystems, to your the Process Commander database to update the data schema. See your Upgrade Guide for more information.

## Running the Install and Upgrade Scripts

After entering the settings in setupDatabase.properties, run the appropriate install or upgrade script for your platform:

- **install.bat** for Windows, **install.ksh** for Unix systems
- **upgrade.bat** for Windows, **upgrade.ksh** for Unix systems

You can pass up to six arguments to either script to override the properties set in the **setupDatabase.properties** file. These arguments are required for any of the properties not defined in the properties file:

- driverClass
- driverJAR
- dbType
- dbURL
- dbUser
- dbPassword

Enter each argument on the command-line followed by the value as described above. Separate each argument with a space. For example:

```
./install.ksh --driverClass oracle.jdbc.OracleDriver  --driverJAR D:\ojdbc6.jar
--dbType oracledate --dbURL jdbc:oracle:thin:@dbServer:1521:prpc_database --
dbUser username --dbPassword password
```

The scripts run an Ant project which installs or upgrades the Process Commander target database as specified. A log of the transactions is written to the command-line console and to a file in the **scripts\logs** directory.

## Advanced Properties

You should not need to set the following properties under normal use. However, they may be useful in certain circumstances.

- **pegarules.config**
  Path to prconfig.xml file
- **prbootstrap.config**
  Path to a prbootstrap.properties file

  The Install and upgrade scripts create their own configuration files based on the settings you specify. Set these two properties if you want the scripts to instead use configuration files you have created. For details on these files see your installation or upgrade guide.

**Notes:**

If you wish to use configuration files, you must create both files and set the corresponding properties. If either of the properties is not set, the script will generate both configuration files.

You must always set the "pega.jdbc.driver.jar" property in setupDatabase.properties. Neither prconfig.xml or prbootstrap.properties contains the location of the DB driver file.

- **bypass.load.engine.classes**

  If set to any value, the Rules Engine classes are not loaded into the database. Use this property if you want to load rules without changing the engine classes.

- **bypass.load.rules**

  If set, the rules are not loaded into the database. Use this property if you want to load the engine classes without changing the rules.

## Other Environment Properties

| Ant Property | Default value | Notes |
|---|---|---|
| jvm.arg.mem.max | -Xmx1024m | Maximum memory allocated to Java tools which are run by this script. |
| pega.install.root | ${basedir} | Path to the scripts directory of the distribution image |
| pega.archives | ${pega.install.root}/../archives/ | Path to the PRPC application archives in the distribution image, like prweb.war. |
| pega.rules | ${pega.install.root}/../rules/ | Path to the PARs (Pega archives) which will be loaded |
| pega.schemas | ${pega.install.root}/../schema/ | Path to a directory named "DDL", containing vendor specific content to be loaded. Each vendor directory must contain a "statements" subdirectory. |