**SOS Berlin setup in Windows/Linux machine**

**Hardware and Software Requirements**

* The hardware requirements to be met before installing JobScheduler are less dependent on JobScheduler itself than on the following factors:
* the operating system,
* the database(s) to be used,
* the jobs that are to be run and how they are to be run (in parallel, serially),
* the software that is controlled by the jobs executed by JobScheduler,
* the memory and data management procedures.
* The minimum requirements for Windows operating systems running on 32-bit hardware are a 1 GHz processor, 2 GB RAM and 16 GB disk space and a 1 GHz processor, 2 GB RAM and 32 GB disk space for 64 bit hardware.
* The same applies to Linux systems, although we run the JobScheduler on Linux VMs with only 512 MB RAM.
* However, this system could also be easily brought to its knees when a sufficient number jobs are executed in parallel. This would be particularly important when, for example, each job was starting its own Java Virtual Machine (JVM).

.

JobScheduler itself doesn't need any dedicated hardware. JobScheduler can be installed and operated on the same server where the database or application is being hosted. Ideally JobScheduler does not have very high resource requirements, i.e. RAM, CPU but it depends on the number and type of jobs that JobScheduler is executing. If JobScheduler is configured with pre-processing and post-processing jobs then JobScheduler will need an extra JVM being loaded per job.

We install JS7 Controller and Cockpit on Linux machine with headless mode.

**For JS7 Controller**

Introduction

* The Controller can be installed on Linux systems without use of an installer by extracting a .tar.gz archive.
* The Controller can be installed on Windows systems
  + without use of an installer by extracting a .zip archive. This allows a portable installation that requires no administrative privileges or
  + by running the Controller installer in headless mode or
  + by running the graphical installer see [JS7 - Controller - Using the Windows Graphical Installer](https://kb.sos-berlin.com/display/PKB/JS7+-+Controller+-+Installation+Using+the+Windows+Graphical+Installer).

Link for Windows Graphical Installer:- <https://kb.sos-berlin.com/display/PKB/JS7+-+Controller+-+Installation+Using+the+Windows+Graphical+Installer>

Download

* Download the Controller archive for the target system from the [JS7 - Download](https://kb.sos-berlin.com/display/PKB/JS7+-+Download) page.
  + For Unix systems:
    - A .tar.gz archive is available that is extracted to create the Controller's directory structure (see below).
      * js7\_controller\_unix.<release>.tar.gz
* For Windows systems:
  + A .zip archive is available that will create the Controller's directory structure when it is extracted (see below).
    - js7\_controller\_windows.<release>.zip
  + A .zip archive including the installer is available that can be used to run the installer in headless mode.
    - js7\_controller\_windows\_installer.<release>.zip

Download JS7 from below given link URL.

Download Agent,Contrller and Cockpit.

Link:- <https://kb.sos-berlin.com/display/PKB/JS7+-+Download>

Link for [JS7 - Controller - Installation Using the Windows Graphical Installer](https://kb.sos-berlin.com/display/PKB/JS7+-+Controller+-+Installation+Using+the+Windows+Graphical+Installer):- <https://kb.sos-berlin.com/display/PKB/JS7+-+Controller+-+Installation+Using+the+Windows+Graphical+Installer>

Link for [JS7 - Controller - Headless Installation on Linux/Windows](https://kb.sos-berlin.com/pages/viewpage.action?pageId=62958081):- <https://kb.sos-berlin.com/pages/viewpage.action?pageId=62958081>

**Create Installation path in Linux**

Path:- /opt/sos-berlin.com/js7

Step 1:-Download js7\_controller\_unix.<release>.tar.gz

Link:- <https://kb.sos-berlin.com/display/PKB/JS7+-+Download>

# Step 2:- Installation Directory:

# The default installation directory for the Controller is:

# /opt/sos-berlin.com/js7/controller/<controller-id> for Unix systems,

# C:\Program Files\sos-berlin.com\js7\controller\<controller-id> for Windows systems.

# Configuration Directory

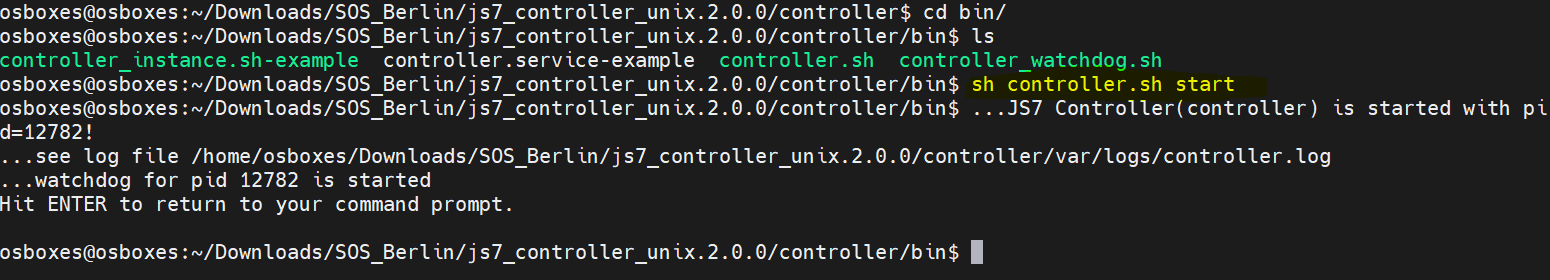
# The default configuration directory for the Controller is:

# /home/sos/sos-berlin.com/js7/controller/<controller-id> for Unix systems,

# C:\ProgramData\sos-berlin.com\js7\controller\<controller-id> for Windows systems.

Step3:- After extracting zip/tar file go to Controller-->bin

Open terminal and run controller.sh file for linux and controller.cmd file for windows with start option.



**For JS7 Cockpit**

Introduction

* The JOC Cockpit can be installed on Linux and Windows systems without use of a graphical installer by running the JOC Cockpit installer in headless mode, for use of a graphical installer see [JS7 - JOC Cockpit - Installation Using the Linux/Windows Graphical Installer](https://kb.sos-berlin.com/pages/viewpage.action?pageId=62959286)

Prerequisites

* The following requirements have to be met before JOC Cockpit can be installed and operated:
* A Java Runtime Environment in version 1.8 or newer is required. For details see [Which Java versions is JobScheduler available for?](https://kb.sos-berlin.com/pages/viewpage.action?pageId=19857566)
* Technically the JOC Cockpit can be operated in any Unix environment. However, the [JS7 - Platforms](https://kb.sos-berlin.com/display/PKB/JS7+-+Platforms) limit support for JOC Cockpit operation to Linux.
* JOC Cockpit requires access to a database to store inventory and status information, and workflow related logs. See the [JS7 - Database](https://kb.sos-berlin.com/display/PKB/JS7+-+Database) article for further information
  + A schema has to be created in the database prior to installation.
  + A user account has to be created that is granted full privileges for the schema.
  + If you intend to use different accounts and privilege sets for setup and for operation of JOC Cockpit then consider switching Hibernate configuration files after setup (see below) to use an account with limited privileges to select, insert, update, delete and execute permissions for schema objects.
* Installation of JOC Cockpit requires administrative privileges in the following situations:
* for Unix systems:
  + - Root permissions are required if JOC Cockpit is to be installed including the Jetty servlet container.
    - To install JOC Cockpit without root permissions the setup script should be executed with the -u argument, otherwise the sudo prompt will ask for the root password.
* for Windows systems:
* The setup requires administrative permissions.

An X-Server is required if the installer is executed on Unix systems in graphical mode. If an X-Server is not installed, then consider using [JS7 - JOC Cockpit - Headless Installation on Linux/Windows](https://kb.sos-berlin.com/pages/viewpage.action?pageId=62959254).

Download

* Download the JOC Cockpit installer archive for your target environment from the [JS7 - Download](https://kb.sos-berlin.com/display/PKB/JS7+-+Download) page.
  + For Unix systems:
    - A .tar.gz archive including the installer is available that can be used to run the installer.
      * js7\_joc\_linux.<release>.tar.gz
* For Windows systems:
  + - A .zip archive including the installer is available that can be used to run the installer.
      * js7\_joc\_windows.<release>.zip
* The .tar.gz/.zip archives can be used to install JOC Cockpit with the graphical installer or to carry out headless installation on both 32 bit and 64 bit operating systems.

# Link for [JS7 - JOC Cockpit - Installation Using the Linux/Windows Graphical Installer](https://kb.sos-berlin.com/pages/viewpage.action?pageId=62959286):- <https://kb.sos-berlin.com/pages/viewpage.action?pageId=62959286>

# Link for [JS7 - JOC Cockpit - Headless Installation on Linux/Windows](https://kb.sos-berlin.com/pages/viewpage.action?pageId=62959254):- <https://kb.sos-berlin.com/pages/viewpage.action?pageId=62959254>

Step 1:-Downloadjs7\_joc\_linux.<release>.tar.gz

Link:- <https://kb.sos-berlin.com/display/PKB/JS7+-+Download>

Step 2:- Default installation directory

Extract the JOC Cockpit archive to a suitable directory such as /home/<user-account> on Unix systems or C:\user\<user-account> on Windows systems.

Step3:- Install PostgreSQL on linux and create Database as Konga, User and role as Konga, Password as Konga.

Download Postgresql for linux using below link

Link:- https://www.postgresql.org/download/linux/ubuntu/

Step5 :- for database connection, download JDBC Driver file (\*.jar) for postgresql using below link

Link:-https://jdbc.postgresql.org/download.html

Step4:- Open and update joc\_install.xml file for DB connection and path configuration

Sample joc\_install.xml file for reference.



<?xml version="1.0" encoding="UTF-8" standalone="no"?>

<!--

XML configuration file for JOC Cockpit setup

The JS7 JOC Cockpit is available with a dual license model:

- GNU GPL v3.0 License, see https://www.gnu.org/licenses/gpl-3.0.en.html

- JS7 Commercial License, see license.txt

The setup asks you for the desired license model,

see below <entry key="licenseOption" .../>

If you run the installer and do not specify a commercial license key

then at the same time you accept the terms of the

license agreement under the GNU GPL v3.0 License.

-->

<AutomatedInstallation langpack="eng">

<com.izforge.izpack.panels.UserInputPanel id="home">

<userInput/>

</com.izforge.izpack.panels.UserInputPanel>

<com.izforge.izpack.panels.UserInputPanel id="licenses">

<userInput>

<!-- Select the license model (GPL or Commercial) -->

<entry key="licenseOption" value="GPL"/>

<!-- If you selected GPL as license model than the 'licenseFile' must be empty.

Otherwise please enter the path to the license file if available.

It is also possible to add the license file later. -->

<entry key="licenseFile" value=""/>

</userInput>

</com.izforge.izpack.panels.UserInputPanel>

<com.izforge.izpack.panels.HTMLLicencePanel id="gpl\_license"/>

<com.izforge.izpack.panels.HTMLLicencePanel id="commercial\_license"/>

<com.izforge.izpack.panels.TargetPanel id="target">

<!-- SELECT THE INSTALLATION PATH

The path must be absolute!

For example:

/opt/sos-berlin.com/js7/joc on Unix

C:\Program Files\sos-berlin.com\js7\joc on Windows -->

<installpath>/opt/sos-berlin.com/js7/joc</installpath>

</com.izforge.izpack.panels.TargetPanel>

<com.izforge.izpack.panels.UserInputPanel id="jetty">

<userInput>

<!-- JOC Cockpit requires a servlet container such as Jetty.

If a servlet container is already installed then you can use it.

Otherwise a Jetty will be installed if withJettyInstall=yes.

You need root permissions to install JOC Cockpit with Jetty. -->

<entry key="withJettyInstall" value="yes"/>

<entry key="jettyPort" value="4446"/>

<!-- Specify the name of the Windows Service or Unix Daemon (default: joc).

Required only for multiple instances of JOC Cockpit on the same server.

The name has to be unique per server. This entry is deactivated by a comment because it

MUST NOT BE MODIFIED DURING RE-INSTALLATION! -->

<!--

<entry key="jettyServiceName" value="joc"/>

-->

<!-- Required for Windows only -->

<entry key="jettyStopPort" value="40446"/>

<!-- Available for Unix only (root permissions required) -->

<entry key="withJocInstallAsDaemon" value="yes"/>

<!-- Enter a user account for running JOC Cockpit (default=current user).

For Unix only (root permissions required)!!! -->

<entry key="runningUser" value=""/>

<!-- Path to Jetty base directory

For example:

/home/[user]/sos-berlin.com/js7/joc on Unix

C:\ProgramData\sos-berlin.com\js7\joc on Windows -->

<entry key="jettyBaseDir" value="/opt/sos-berlin.com/js7/"/>

<!-- Choose 'yes' or 'no' whether the JOC Cockpit's Jetty should be (re)started after installation -->

<entry key="launchJetty" value="yes"/>

<!-- Java options for Jetty -->

<!-- Initial memory pool (-Xms) in MB -->

<entry key="jettyOptionXms" value="128"/>

<!-- Maximum memory pool (-Xmx) in MB -->

<entry key="jettyOptionXmx" value="512"/>

<!-- Thread stack size (-Xss) in KB -->

<entry key="jettyOptionXss" value="4000"/>

<!-- Further Java options -->

<entry key="jettyOptions" value=""/>

</userInput>

</com.izforge.izpack.panels.UserInputPanel>

<com.izforge.izpack.panels.UserInputPanel id="joc">

<userInput>

<!-- JOC Cockpit can be installed in a cluster. Please type a unique title to identify the cluster node,

e.g. 'Primary' or 'Secondary' (default = hostname). Max. length is 30 characters. -->

<entry key="jocTitle" value="joc"/>

<!-- Choose 'yes' if JOC Cockpit is operated as a standby instance in a cluster -->

<entry key="isStandby" value="no"/>

<!-- Security Level for the deployment signing mechanism: possible values are 'LOW', 'MEDIUM' and 'HIGH':

HIGH:

RSA/ECDSA certificates are stored for verification per user,

signing is performed externally outside of JOC Cockpit.

MEDIUM:

RSA/ECDSA private keys and certificates are stored for signing per user,

signing is performed automatically with the user's private key.

LOW:

a single RSA/ECDSA private key and certificate are stored with the root account,

signing is performed automatically with the root account's private key for all users. -->

<entry key="securityLevel" value="LOW"/>

</userInput>

</com.izforge.izpack.panels.UserInputPanel>

<com.izforge.izpack.panels.UserInputPanel id="database">

<userInput>

<!-- Reporting Database Configuration -->

<!-- Database connection settings can be specified with subsequent entries such as

databaseHost, databasePort, ... or by a Hibernate configuration file.

or you choose the embedded H2 database (for evaluation only) without further configuration.

Possible values are 'withoutHibernateFile', 'withHibernateFile' and 'h2'. -->

<entry key="databaseConfigurationMethod" value="withoutHibernateFile"/>

<!-- Choose the database management system.

Supported values are 'mysql' for MySQL, 'oracle' for Oracle,

'mssql' for Microsoft SQL Server, 'pgsql' for PostgreSQL.

Used only if databaseConfigurationMethod=withoutHibernateFile -->

<entry key="databaseDbms" value="pgsql"/>

<!-- Path to a Hibernate configuration file if databaseConfigurationMethod=withHibernateFile -->

<entry key="hibernateConfFile" value=""/>

<!-- You can choose between 'byInstaller', 'byJoc' or 'off' to create the database objects.

Data of an already existing installation remain unchanged.

This entry should be 'off' only when you are certain that all objects have already been created.

'byJoc' creates or updates database object the next time JOC Cockpit is started.

'byInstaller' is recommended. If databaseConfigurationMethod=h2 then the objects will be

created when JOC Cockpit starts and this value will be ignored -->

<entry key="databaseCreateTables" value="byInstaller"/>

</userInput>

</com.izforge.izpack.panels.UserInputPanel>

<com.izforge.izpack.panels.UserInputPanel id="dbconnection">

<userInput>

<!-- Database configuration if databaseConfigurationMethod=withoutHibernateFile -->

<!-- Enter the hostname or ip address of the database host.

This entry can also be used to configure the URL(s) for Oracle RAC databases.

For example:

<entry key="databaseHost" value="(DESCRIPTION=(ADDRESS\_LIST=(LOAD\_BALANCE=OFF)(FAILOVER=ON)

(ADDRESS=(PROTOCOL=TCP)(HOST=tst-db1.myco.com)(PORT=1604))

(ADDRESS=(PROTOCOL=TCP)(HOST=tst-db2.myco.com)(PORT=1604)))

(CONNECT\_DATA=(SERVICE\_NAME=mydb1.myco.com)(SERVER=DEDICATED)))"/>

The "databaseSchema" and "databasePort" entries should be left empty. -->

<entry key="databaseHost" value="localhost"/>

<!-- Enter the port number for the database instance. Default ports are for MySQL 3306,

Oracle 1521, Microsoft SQL Server 1433, PostgreSQL 5432. -->

<entry key="databasePort" value="5432"/>

<!-- Enter the database schema -->

<entry key="databaseSchema" value="konga"/>

<!-- Enter the user account for database access -->

<entry key="databaseUser" value="konga"/>

<!-- Enter the password for database access -->

<entry key="databasePassword" value="konga"/>

</userInput>

</com.izforge.izpack.panels.UserInputPanel>

<com.izforge.izpack.panels.UserInputPanel id="jdbc">

<userInput>

<!-- Database configuration if databaseConfigurationMethod != h2 -->

<!-- You can specify an external JDBC Driver, in this case set internalConnector=no

For license reasons MySQL and Microsoft SQL Server JDBC drivers are not included.

Alternatively you can use the MariaDB JDBC Driver for MySQL that is included.

For Microsoft SQL Server you have to specify an external JDBC Driver.

An Oracle ojdbc8 JDBC driver is included. -->

<!-- You can choose between 'yes' or 'no' for using the internal JDBC Driver -->

<entry key="internalConnector" value="yes"/>

<!-- Select the path to the JDBC Driver file (\*.jar) -->

<entry key="connector" value="/home/osboxes/Downloads/SOS\_Berlin/postgresql-42.2.24.jar"/>

</userInput>

</com.izforge.izpack.panels.UserInputPanel>

<com.izforge.izpack.panels.UserInputPanel id="end">

<userInput/>

</com.izforge.izpack.panels.UserInputPanel>

<com.izforge.izpack.panels.InstallPanel id="install"/>

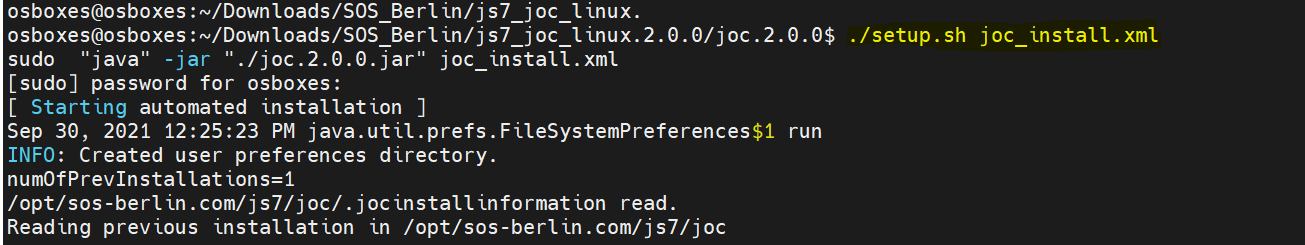
<com.izforge.izpack.panels.ProcessPanel id="process"/>

<com.izforge.izpack.panels.FinishPanel id="finish"/>

</AutomatedInstallation>

Step 5:- Go to js7\_joc\_linux.2.0.0-->joc.2.0.0

Open terminal and run ./setup.sh joc\_install.xml

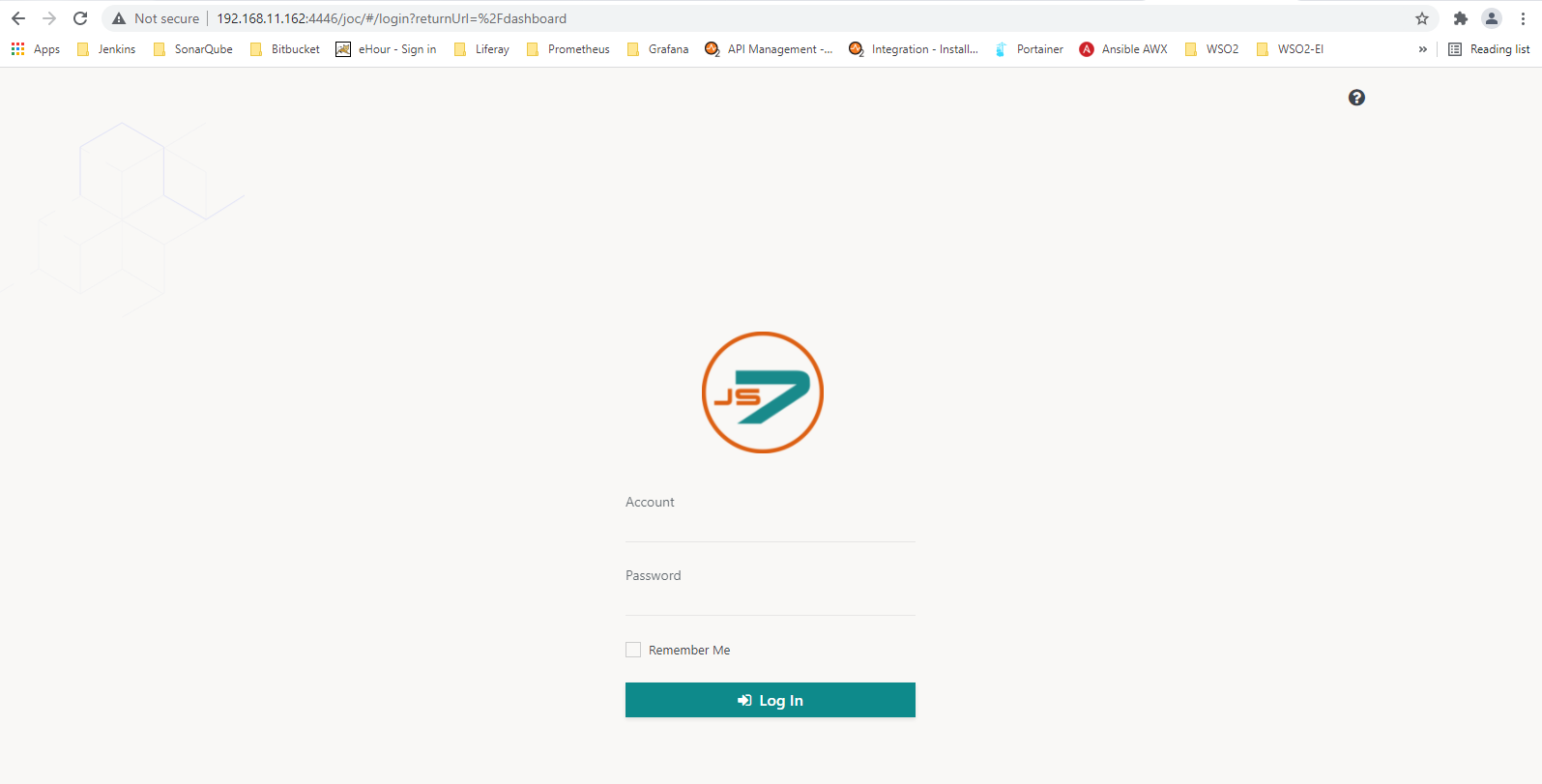


Step 6:- Go to installpath directory (/opt/sos-berlin.com/js7/joc/jetty/bin/) as below and run command as

sh jetty.sh start



Step 7:- Open Browser and hit url with local host/IP with port no 4446



Step 8:- Enter controller url with port no 4444 ,and then enter account as “root”and password as “root”.

