



Kuali People Management (KPME) Installation Guide

Release v. 1.1

1/2/2012



Contents

Copyright and Licensing	i
Software Licensing	i
Contributor Licensing	i
Documentation Licensing	ii
Downloading, Installing, and Configuring	1
Installation Basics and High-Level Overview	1
Technical Requirements	2
Distribution Download	3
Quickstart Setup	4
Embedded Mode	8
Where To Find More KPME Technical Documentation	9
Appendix A: Configuration Parameters	10



Copyright and Licensing

All Kuali Foundation Licensing information is located online at:
<https://test.kuali.org/confluence/display/KULFOUND/Licensing>



Kuali ® is a registered trademark ™ of the Trustees of Indiana University.

The Kuali Foundation has established a set of intellectual property management practices to protect the foundation, its members, and the extended Kuali community.

The Kuali Foundation uses various licenses to distribute software and documentation, to accept regular contributions from individuals and organizations, and to accept large grants of existing software products.

The sections that follow explain Kuali copyright information as it pertains to the following three licensing areas:

- Software Licensing
- Contributor Licensing
- Documentation Licensing

Intellectual Property Contact Information: If you have any questions about Kuali Intellectual property, please contact The Kuali Foundation at licensing@kuali.org.

Software Licensing



Copyright © 2007-2011 The Kuali Foundation. All rights reserved. Kuali People Management is licensed for use pursuant to the Educational Community License, Version 2.0 (<http://www.opensource.org/licenses/ec12.php>). Portions of Kuali People Management copyrighted by other parties, including the parties listed below, and you should see the licenses directory for complete copyright and licensing information. Questions about licensing should be directed to license@kuali.org.

Contributor Licensing

Portions of Kuali People Management were developed by Indiana University, Iowa State University, University of Georgia, and Vivantech Inc.. (<http://www.kuali.org/>).

This product includes software developed by the Apache Software Foundation (<http://www.apache.org>).

This product includes software developed by ANTLR (<http://www.antlr.org/>).

This product includes software developed by the JAX-RPC Project part of Project GlassFish (<https://jax-rpc.dev.java.net/>).

This product includes software developed by the SAAJ Project part of Project GlassFish (<https://saa.dev.java.net/>).

This product includes software developed by Displaytag (<http://displaytag.sourceforge.net/11/>).

This product includes software developed by the JDOM Project (<http://www.jdom.org/>).

This product includes software developed by the Open Symphony Group (<http://www.opensymphony.com/>).



This product includes software developed by the SAXPath Project (<http://www.saxpath.org/>).

This product includes software developed by the JA-SIG Collaborative (<http://www.ja-sig.org/>).

This product includes software licensed under the Apache Software License (<http://www.apache.org/licenses/>).

This product includes software licensed under the Common Development and Distribution License (<http://www.opensource.org/licenses/cddl1.php>).

This product includes software licensed under the Common Public License (<http://www.opensource.org/licenses/cpl1.0.php>).

This product includes software licensed under the Mozilla Public License (<http://www.mozilla.org/MPL/>).

This product includes the Kuali Rice module licensed under the Kuali Foundation ECL ([Rice Acknowledgments](#)).

Portions Copyright (c) 2005 Envoi Solutions LLC. All Rights Reserved. (<http://xfire.codehaus.org/>)

Portions Copyright (c) 2000-2006 The Legion of the Bouncy Castle. All Rights Reserved. (<http://www.bouncycastle.org>)

Portions Copyright (c) 2001-2005 MetaStuff, Ltd. All Rights Reserved. (<http://www.dom4j.org/>)

Portions Copyright (c) 2003-2007 Joe Walnes and the XStream Committers. All Rights Reserved. (<http://xstream.codehaus.org/>)

Portions Copyright (c) 2003-2006 The Werken Company. All Rights Reserved. (<http://jaxen.codehaus.org/>)

Portions Copyright (c) 1998-2003 World Wide Web Consortium (Massachusetts Institute of Technology, European Research Consortium for Informatics and Mathematics, Keio University). All Rights Reserved. This work is distributed under the W3C® Software License in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

Documentation Licensing



Copyright © 2011 by The Kuali Foundation. Some rights reserved.

Kuali People Management User Documentation by the [Kuali Foundation](#) is licensed under a [Creative Commons Attribution-Share Alike 3.0 United States License](#). Permissions beyond the scope of this license may be available at <http://www.kuali.org>.

Creative Commons License Deed



Attribution-Share Alike 3.0 United States

You are free:



to Share – to copy, distribute, and display the work



to Remix – to make derivative works

Under the following conditions:



Attribution. You must attribute this work to the [Kuali Foundation](#).



Share Alike. If you alter, transform, or build upon this work, you may distribute the resulting work only under the same, similar or a compatible license.



Permissions beyond the scope of this public license are available at <http://www.kuali.org>.

- For any reuse or distribution, you must make clear to others the license terms of this work. The best way to do this is with a link to <http://www.kuali.org>.
- Any of the above conditions can be waived if you get permission from the [Kuali Foundation](#), the copyright holder.
- Apart from the remix rights granted under this license, nothing in this license impairs or restricts the author's moral rights.

Disclaimer

Your fair use and other rights are in no way affected by the above.

The Commons Deed is not a license. It is simply a summary and handy reference for understanding the [Legal Code](#) (the full license) — it is a human-readable expression of some of its key terms. Think of it as the user-friendly interface to the Legal Code beneath. This Deed itself has no legal value, and its contents do not appear in the actual license. Creative Commons is not a law firm and does not provide legal services. Distributing of, displaying of, or linking to this Commons Deed does not create an attorney-client relationship.

The information in this document is subject to change without notice. It has been obtained from sources believed to be reliable. Although the authors and the Kuali Foundation have made every effort to ensure the accuracy of this document, neither the authors nor the publisher assumes any liability or responsibility for any inaccuracy or omissions contained therein, or for any loss or damage arising from the information presented.

If you discover any issues with the documentation, please report your findings, in writing, to The Kuali Foundation. The Kuali Foundation does not warrant that this document is error-free.

All other products or company names represented in this document are not to be viewed as endorsements by either the authors or The Kuali Foundation, and may be trademarks for their respective owners.



Downloading, Installing, and Configuring

Installation Basics and High-Level Overview

This topic describes how to install Kuali People Management for Enterprise (KPME) on a single server. The KPME application binaries and source code are included as part of the installation distribution.

The KPME application assists in the accessibility, administration and support of people management functions. There are many components that are working together to configure and install KPME. The core system is known as Kuali Rice (KR), which is a set of integrated middleware products that allows KPME to interface with technical modules running from a web server. Kuali Rice facilitates enterprise workflow functionality and provides customizable and configurable user interfaces that have a clean and universal look and feel. Kuali Rice contains five core modules of which three are actively used in KPME. Kuali Enterprise Workflow (KEW) monitors document routing and approval while automating electronic processes and transactions across the enterprise. The Kuali Nervous System (KNS) alternately handles functionality common to many modules and can be thought of as a framework of reusable components providing an abstraction layer for developers to easily integrate with other Kuali Rice components. The Kuali Service Bus (KSB) assists with scheduling for batch processing.

KPME scales depending upon your environment needs and therefore allows for various architectural choices to be implemented during installation. This topic covers the essential architectural components that are vital to the overall operation of the system. KPME handles data storage using a database to maintain records. Care must be taken to ensure that the database remains accessible during times when KPME is active. The KPME application server manages data for presentation through a web browser.



Technical Requirements

KPME is versatile in its ability to deploy on numerous platforms but has only been tested on a select set of platforms. This section describes specific requirements to install KPME.

System Server

This section outlines the minimum **server requirements** for KPME.

- **Hardware:** Recommended minimum hardware for the web application is 2 cores of a recent CPU architecture along with 2 gig of reserved RAM. Minimum hardware and settings for the database should be based on vendor recommendations, but there should be high bandwidth with low latency between the web application and database as this is a database intensive application.
- **Operating Systems:** Any OS which supports either MySQL or Oracle databases should work, including Solaris, but it has been run on Linux (Ubuntu and RedHat distributions) only.
- **Web Application Platform:** The Web application platform can be any fully compliant modern J2EE container. Testing has been done with Jetty 6.1, Tomcat 5.5, and Tomcat 6.
- **Heap Space:** The Webapp generally requires 1 gig of heap space (-Xmx1g) and 256 meg of perm gen space (-XX:MaxPermSize=256m) to start and run. For production use, at least 2 gig of heap and 512 meg of perm gen space are recommended.
- **Other Recommended Settings:** Other recommended settings are -server (to improve resource utilization for long-running tasks) and -XX:+UseConcMarkSweepGC (to improve garbage collection performance).

Supporting Application Software

This section outlines the minimum **software requirements** for KPME.

- Oracle Java Development Kit (JDK 1.5.x minimum, JDK 1.6 recommended)
- Web Application Server (Apache 2.x) – as needed, depending on network infrastructure
- Servlet container (Apache Tomcat 5.5.x)
- Web Browser (Firefox 1.5.x, Internet Explorer 6.x or Safari 2.0.4.x)
- Apache Ant 1.7.x
- Maven 2.0.x
- Oracle Database 10g Release 2 (10.2.0.x)
- Oracle Database 10g Release 2 (10.2.0.x) JDBC Driver
- MySQL 5.x
- MySQL Connector/J



Database

This section outlines the minimum database **requirements** for KPME.

KPME has been tested against Oracle and MySQL. Supported versions of database products that KPME has been tested with are as follows:

- Oracle10g/11g: JDBC drivers for Oracle can be obtained from <http://www.oracle.com/index.html>
- MySQL: JDBC drivers for MySQL can be obtained from www.mysql.com

KPME requires a User Account with access **permissions** to the database to perform the following:

- Connect to the database
- Create, Alter and Drop Tables, Triggers, Views, Procedures and Sequences
- Insert, Update and Delete data in the database tables

KPME recommends the following database **settings**:

- For MySQL running on a case-sensitive OS (i.e., Linux) it is recommended to use settings to force all table names to lowercase (see other MySQL recommended settings in the Rice documentation).

Distribution Download

To download KPME navigate to the following site and fill out the download form appropriately:

<http://www.kuali.org/download-form>

The KPME distribution is available from the Kuali Foundation at

<https://svn.kuali.org/repos/kpme/tags/time-and-attendance-1.0/>

The KPME download contains:

- KPME source code
- Database scripts – Full install
- Binary distribution
- Installation Guide (this document)
- Project licenses
- List of known bugs

Extract the ZIP file that you downloaded to the folder of your choice.

Quickstart Setup

Use the following installation instructions to guide you through the baseline out-of-the-box setup.

 **Before You Begin** - The following are required:

- Java 1.6
- Servlet 2.4 / JSP 2.0 Compatible Servlet Container such as Apache Tomcat 5.5
- Oracle database instance or MySQL database instance

	<p>To set up KPME (and Rice) database:</p> <p>KPME database release contains all SQL scripts needed to install a new schema for version 1.1 database using Rice embedded mode with the database objects (tables, constraints, bootstrap data) for the KPME application.</p> <p>If installing in an embedded Rice environment you will need to verify this is version 1.0.3.</p>
	<p>Installation Steps for Oracle</p> <hr/> <p> Installation has been tested with sqlplus and SQL Developer. Other tools may require modification of the scripts.</p> <hr/>
1.	Create Oracle username of less than 8 characters
2.	<p>Make sure Oracle user has following privileges:</p> <ul style="list-style-type: none"> * DEFAULT TABLESPACE set to <Users Tablespace> (the intended location where the schema database structures and base bootstrap data are stored). * QUOTA UNLIMITED ON <Users Tablespace> * CREATE SESSION * CREATE SYNONYM * CREATE PROCEDURE * CREATE TRIGGER * CREATE TABLE * CREATE TYPE * CREATE VIEW * CREATE SEQUENCE <hr/> <p> A user's DEFAULT TABLESPACE is set with the CREATE USER statement or ALTER USER statement. The TABLESPACE should not be the SYSTEM tablespace.</p> <hr/> <p> A New install will COMPLETELY clear data from any existing KPME tables in this schema!</p> <hr/>
	<p>Installation Steps for MySQL</p> <hr/> <p> Installation has been tested with mysql client. Other tools may require modification of the scripts.</p> <hr/>
1.	Set the following settings in MySQL:

	<p>max_allowed_packet=1M</p> <p>transaction-isolation=READ-COMMITTED</p> <p>lower_case_table_names=1</p> <p>max_connections=1000</p>
2.	Create MySQL username of less than 8 characters.
3.	Create MySQL default schema for username with the default character set of UTF8. If a different character set is desired, the ddl scripts will need to be updated to the new character set.
4.	<p>Make sure MySQL user has following privileges on the schema</p> <ul style="list-style-type: none"> * Select * Insert * Update * Delete * Create * Drop * Index * Alter * Create_view * Create_routine * Alter_routine * Create_tmp_table * Lock_tables <hr/> <p> Be sure the schema is empty before a NEW installation.</p>
End of activity.	

	To populate data into KPME (and Rice) database:
	Installation Steps for Both Oracle and MySQL
1.	Run the appropriate install script in src/main/config/sql for the appropriate schema.
2.	(Optional) Install demonstration data. Convenience install scripts for demonstration data can be found in the src/main/config/sql/testdrive directory . The demonstration data is only guaranteed to install cleanly into a new KPME install. If you have previous test data or demonstration data then you will encounter errors, which may or may not affect functionality.

 Demonstration data is for testing functionality of the software only and should not be included in an actual implementation. Take a backup of your database before applying demonstration data or be prepared to reinstall from scratch before implementation.
--



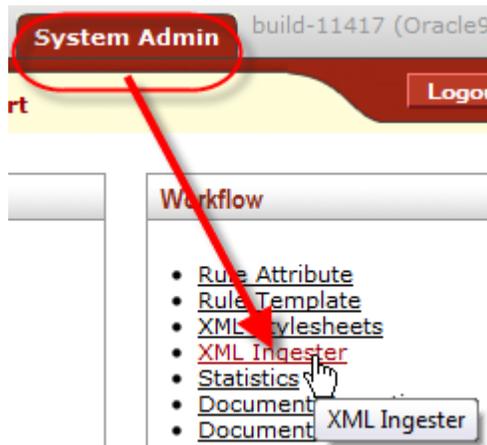
To set up KPME Application:

1. Be sure database is setup and populated with KPME and Rice data (as instructed above).
2. Place src/main/config/local/kuali/* in the following directory: {Userhome}/kuali
 - To use a different location, open tk-dev.war and create a "config.dir" parameter in /WEB-INF/classes/META-INF/tk-config.xml.
 - You SHOULD also be able to pass a java parameter of config.dir to your j2ee container (not guaranteed to work)



This file also contains the "build.environment" parameter referenced in the external tk-config.xml file as "environment"

3. Edit parameters in the external tk-config.xml (that you just placed in step #2) as desired.
 - You will at least need to fill in the following parameters with your database connection information: datasource.url, datasource.username, datasource.password.
 - You will also need to change application.host to your server's host name (i.e. kuali.yourinstitution.edu).
 - Other commonly changed parameters are: application.http.scheme, http.port and app.context.name (be sure app.context.name matches your deployed webapp -- normally the name of your war file)
 - If you are using SSL, be sure that application.http.scheme and http.port reflect this.
 - If you are using MySQL, see the MySQL changes at the end of this guide.
4. **For Oracle:** Obtain ojdbc14.jar from the Oracle website (<http://www.oracle.com>).
For MySQL: Maven should install this for you.
Example: For Tomcat 5.5, copy ojdbc14.jar into {tomcat install dir}/common/lib.
5. Deploy time-and-attendance1.1.war to your servlet container (see user guide for your particular servlet container).
6. The KPME application should now be accessible at: {application.host}/{app.context.name} (as specified in the external tk-config.xml -- app.context.name is tk-dev, if not overridden).
7. Install Standalone Kuali Rice using information at <http://www.kuali.org>
8. Login as admin in your Kuali Rice instance. Ingest the KEW documents.
 - Click the System Admin tab and then select XML Ingestor from the Workflow pane.



- Ingest all files in KPME/src/main/config/workflow/*
- Ingestion can take several minutes per zip file (up to around 10 min), during which time the activity indicator on your browser will just stay animated.

End of activity.



Configuration Parameters



Tip: Refer to “[Appendix A: Configuration Parameters](#)” on page 12 within this documentation set for supplementary information that will prove helpful by providing default values and handy tips for both KPME-specific and Rice parameters.

Embedded Mode

Embedded mode allows you to run KPME with a central Rice server. This has the benefit of a central Doc Search, Action List, and KIM maintenance.

KPME is only setup to run in this state currently. If you wish to run bundled some information can be found in the Kuali Rice documentenation.



For a more detailed description of embedded mode, see the Rice documentation [<https://test.kuali.org/confluence/display/KULRICE/Documentation>].



Where To Find More KPME Technical Documentation

The following is a list of links to other sources of information related to KPME installation, and references to additional online resources where you can find more information that may prove to be helpful for installation activities:

Kuali Resources

- **KPME Technical Documentation:** <https://wiki.kuali.org/display/KPMEDOC/Home>
- **Kuali Rice Documentation:** <https://test.kuali.org/confluence/display/KULRICE/Documentation>
- **Kuali Architecture and Development Standards:** <http://kuali.org/files/pdf/KualiStandards.pdf>
- **KPME Test Drive:** <http://testdrive.kpme.kuali.org>

Tomcat Resources

- **The Apache Tomcat 5.5 Servlet/JSP Container:** <http://tomcat.apache.org/tomcat-5.5-doc/index.html>
- **The Apache Jakarta Project:** <http://jakarta.apache.org/>

Eclipse Resources

- **Eclipse Documentation:** <http://www.eclipse.org/documentation/>

Oracle Resources

- **Oracle Database Express Edition Getting Started Guide:**
http://download.oracle.com/docs/cd/B25329_01/doc/admin.102/b25610/toc.htm
- **Oracle Express Edition Tutorial:** <http://st-curriculum.oracle.com/tutorial/DBXETutorial/index.htm>



Appendix A: Configuration Parameters

The following information is provided for **each parameter**:

- **Name** – The parameter name, for example, ‘build.properties’
- **Purpose** – A textual description of the parameter’s purpose
- **Default Value** – The pre-configured value prior to customization
- **Notes** – Information that aids in the understanding of proper usage

build.environment

Purpose: The environment the application is running in (ex: dev, prd, tst).

Default Value: dev

Notes: Since the Rice param environment=\${build.environment}, this parameter is indirectly used by rice to construct urls and resolve config files.

build.version

Purpose: The version number displayed on the application header.

Default Value: NO VERSION

Notes: Since the rice param version=\${build.version}, this parameter is indirectly used by rice for display purposes.

dev.mode

Purpose: Used when setting the runmode.

Default Value: FALSE

Notes: set to false when running in embedded mode. See rice documentation for more details.

rice.app.url

Purpose: The url to the rice server.

Default Value: \${workflow.url}

Notes: For embedded mode this will point to a standalone server.

Notes: (see rice.struts.message.resources)

in the org.kuali.rice.ksb.messaging.ApplicationResources and ApplicationResources (kpme's message bundle) kpme's message will take precedent.

datasource.url

Purpose: the url of the client (or client & rice) database

Default Value: jdbc:oracle:thin:@127.0.0.1:1521:KUALI

Notes: in embedded mode this will be the KPME database only.

datasource.username

Purpose: username of the client (or client & rice) database

Default Value: KRADEV



Notes: (see datasource.url)

datasource.password

Purpose: password of the client (or client & rice) database

Default Value: ask your team

Notes: (see datasource.url)

datasource.platform

Purpose: platform of the client (or client & rice) database

Default Value: Oracle9i

Notes: (see datasource.url)

server.datasource.url

Purpose: rice server database property

Default Value: \${kr.datasource.url}

Notes: in embedded mode this will be the Rice database only.

server.datasource.username

Purpose: rice server database property

Default Value: \${kr.datasource.username}

Notes: (see server.datasource.url)

server.datasource.password

Purpose: rice server database property

Default Value: \${kr.datasource.password}

Notes: (see server.datasource.url)

server.datasource.ojb.platform

Purpose: rice server database property

Default Value: \${kr.datasource.ojb.platform}

Notes: (see server.datasource.url)

server.datasource.platform

Purpose: rice server database property

Default Value: \${kr.datasource.platform}

Notes: (see server.datasource.url)

server.datasource.driver.name

Purpose: rice server database property

Default Value: \${kr.datasource.driver.name}

Notes: (see server.datasource.url)



server.datasource.pool.validationQuery

Purpose: rice server database property

Default Value: `${datasource.pool.validationQuery}`

Notes: (see server.datasource.url)

server.datasource.pool.maxWait

Purpose: rice server database property

Default Value: `${datasource.pool.maxWait}`

Notes: (see server.datasource.url)

server.datasource.pool.minSize

Purpose: rice server database property

Default Value: `${datasource.pool.minSize}`

Notes: (see server.datasource.url)

server.datasource.pool.maxSize

Purpose: rice server database property

Default Value: `${datasource.pool.maxSize}`

Notes: (see server.datasource.url)

server.datasource.pool.size

Purpose: rice server database property

Default Value: `${datasource.pool.size}`

Notes: (see server.datasource.url)

server.datasource.connectionTimeout

Purpose: rice server database property

Default Value: `${datasource.connectionTimeout}`

Notes: (see server.datasource.url)

keystore.alias

Purpose: The alias for the rice keystore file.

Default Value: `onestartshareservices-devandtst`

Notes: required for kpme client application to communicate with a standalone rice server (embedded mode)

keystore.file

Purpose: The path & file name to the rice keystore file.

Default Value: `ku1`

Notes: (see keystore.alias)



keystore.password

Purpose: The password for the rice keystore file.

Default Value: kulpass

Notes: (see keystore.alias)