

Volume

1



Technical Documentation for UNIX[®]/Linux[®]

KETL[®] Installation Guide 2.1

TECHNICAL DOCUMENTATION FOR UNIX®/LINUX®

KETL® Installation Guide 2.1

© 1999 – 2008. All rights reserved worldwide.
Kinetic Networks, Inc.
33 New Montgomery Street • Suite 1200
Phone 415.358.5100 • Fax 415.358.5101

KETL is a registered trademark of Kinetic Networks, Inc.
UNIX is a registered trademark of The Open Group. Linux is a registered trademark of Linus Torvalds.
Windows is a registered trademark of Microsoft Corporation in the United States and other countries.

Table of Contents

1	INTRODUCTION			
	Scope	1	Creating the Metadata	10
	Audience	1	Oracle	10
	Related Documents	1	PostgreSQL	10
			Configuring the Initial Environment	11
			Testing the Environment	11
2	REQUIREMENTS			
	Java	2	4	TROUBLESHOOTING
	Database	2	Connectivity Issues	15
	Database Drivers	3	Missing Database Drivers	15
	MySQL JDBC Connectivity	3	Missing Path	15
	Oracle JDBC Connectivity	3	Invalid Connect String	16
	Procedural Language	4	Failure to Start the KETL Server	17
	Complex Transformations	4	Disconnected KETL Server	17
	Optional Items	5	Incorrect Environment Settings	18
	Enhanced Logging	5	Incorrect Permissions	18
	Web Services/SOAP Protocols	5	Failure to Stop the KETL Server	18
	XML Parsing	6	Incorrect server ID	18
	PostgreSQL Thin JDBC Drivers	6	Invalid Privileges	19
			Failure to Create Log Files	19
			Failure to Create Metadata	19
			Missing Procedural Language	19
3	PROCEDURES		PostgreSQL Bug	20
	Creating a Dedicated Operating System		Missing Jar Files	21
	User and Group	8	tools.jar	21
	Installing KETL	8		



Introduction

Kinetic Networks[®], Inc. (KNI) has developed a flexible and robust Java application to extract, transform, and load data. This application, named KETL[®], allows management of complex manipulation of data while leveraging an open source data integration platform. KETL is a metadata-driven solution that is scalable and platform-independent. The engine is built upon an open, multi-threaded, XML-based architecture. As such, complex ETL transformations can be executed quickly and efficiently.

Scope

This document explains how to install KETL onto your UNIX[®] or Linux[®] system. This document does not explain how to use KETL, only how to install the application.

To install KETL onto the Windows[®] platform, refer to the following document:

- KETL Installation Guide for Windows

Audience

This document is for the operations and administrative staff, typically those who install and configure enterprise software.

Related Documents

After installation of KETL, refer to the following documents as needed:

- KETL System Administration Guide
- KETL Developer's Guide

Requirements

To install KETL, you need a Java run-time environment and a transactional database to store the metadata. This section describes the following system requirements:

- [Java](#)
- [Database](#)
- [Database Drivers](#)
- [Procedural Language](#)
- [Complex Transformations](#)

This section also describes [optional items](#) for your installation.

Java

KETL runs under any operating system using Java 1.5 (or higher).

Database

You need a relational database to store the KETL metadata. In general, you can use any relational database that supports row-level locking. KETL, however, has been tested on the following databases:

Database	Version
PostgreSQL	8.0 and higher
Oracle	9.1 and higher
MySQL	5.1 or higher using InnoDB for storage

KETL bundles scripts for Oracle and PostgreSQL, if you choose to store the KETL metadata on either of these databases.

Note

This database is for the KETL metadata repository and is not a restriction for either data sources or targets.

Database Drivers

You need thin client JDBC drivers for connection to the appropriate metadata database. Due to licensing restrictions, KETL only provides client drivers for PostgreSQL.

Both MySQL and Oracle require specific jar files to enable JDBC connectivity. You can download the jar file to any directory, but KETL automatically finds all files in \$KETLDIR/lib. If you copy your jar file to a different directory, then be sure to update the \$KETLDIR/conf/Extra.Libraries file with the correct path.

**Important**

After you update your environment variables, be sure to source the file. Otherwise, the system does not recognize that you made any changes.

MySQL JDBC Connectivity

If you use MySQL for the KETL metadata repository, complete the following steps:

1. Go to the following location:

<http://dev.mysql.com/downloads/connector/j/5.0.html>

2. Download the latest jar file.

For example:

`mysql-connector-java-5.0.x-bin.jar`

Where x is the latest version of the driver.

Oracle JDBC Connectivity

If you use Oracle for the KETL metadata repository, complete the following steps:

1. Go to the following location:

http://www.oracle.com/technology/software/tech/java/sqlj_jdbc/index.html

2. Download the following file:

`ojdbc14.jar`

Procedural Language

Support for procedural language in the SQL engine of the metadata repository must be enabled:

Database	Language
PostgreSQL	pgSQL
Oracle	PL/SQL
MySQL	mysql (usually enabled by default in MySQL 5.0 and higher)



Important

The appropriate procedural language must be installed so that you can successfully create the metadata repository.

Complex Transformations

Complex transformations are optional, but KNI recommends that you enable this feature for your KETL installation. Due to licensing restrictions, the jar file for this feature is not included in your KETL distribution. This class library, however, is usually bundled with your Java SDK installation. Using this class allows dynamic compilation of the transformations.

To enable complex transformations, you need the following jar file:

```
tools.jar
```

Complete the following steps:

1. Set your JAVA_HOME environment variable.

For example:

```
JAVA_HOME=/usr/java/jdk1.5.0_10
export JAVA_HOME
```

2. Include JAVA_HOME in the path.

For example:

```
PATH=$PATH:$HOME/bin:$JAVA_HOME/bin:$KETLDIR/bin
export PATH
```

3. Save and exit the file.
4. Run the appropriate source command on your file.

Optional Items

The following features are optional for your KETL installation:

- [Enhanced logging](#)
- [Web services/SOAP protocols](#)
- [XML parsing](#)
- [PostgreSQL thin JDBC drivers](#)

These class libraries have licensing that allows for inclusion in your KETL distribution. For copies of the license agreements, go to the \$KETL/license directory. If you update any of the libraries, then replace the copies in the \$KETLDIR/lib directory.



Important

Check the sources for the most recent versions of the class libraries, as these libraries are constantly being updated.

Enhanced Logging

The following class libraries allow detailed levels of system and error logging:

```
Log4j-1.x.x.jar
Commons-logging-1.x.x
```

For more information, refer to the following site:

```
http://logging.apache.org/log4j/docs/download.html
```

Web Services/SOAP Protocols

The following class libraries contain lightweight protocol for exchanging structured information in a decentralized, distributed environment (such as the Internet):

```
axis.jar
jaxrpc.jar
saaj.jar
```

It is an XML-based protocol that consists of the following parts:

- An envelope that defines a framework for describing what is in a message and how to process it.
- A set of encoding rules for the expressing instances of application-defined data types.
- A convention for representing remote procedure calls and responses.

For more information, refer to the following site:

<http://www.apache.org/dyn/closer.cgi/ws/axis>

XML Parsing

The following libraries are implementations of XSLT 2.0, XPath 2.0, and XQuery 1.0:

saxon8.jar
saxon8-dom.jar
saxon8-xpath.jar

These libraries provide all languages except for schema-aware processing.

For more information, refer to the following site:

<http://saxon.sourceforge.net>

PostgreSQL Thin JDBC Drivers

The following library allows connection to the PostgreSQL database:

postgresql.jar

This library provides the connectivity to PostgreSQL as the repository for the KETL metadata and the ability to use PostgreSQL as either a target or a source for data flows.

For more information, refer to the following site:

<http://jdbc.postgresql.org/download.html>

Procedures

This section describes the following procedures:

- [Creating a Dedicated Operating System User and Group](#)
- [Installing KETL](#)
- [Creating the Metadata](#)
- [Configuring the Initial Environment](#)
- [Testing the Environment](#)

These procedures assume that the KETL server is running on the same machine as the KETL metadata database.

Note

Some procedures may require you to create a new user, group, or directory. If so, these instructions provide a sample name (such as “ketl” or “ketlmd”) which you are free to rename.

Creating a Dedicated Operating System User and Group

While KETL can be installed and operated from any account, for ease of use and security, KNI recommends creating a dedicated user account and group for the operating system.

Complete the following steps:

1. Log in as root or any user with sudo privileges.
2. Create a new user “ketl” with group “ketl.”
3. Create a “ketl” directory on your system.

The examples use the following directory:

```
/usr/local/ketl
```

4. Change the user and group ownership of the new directory.

Depending on your operating system, you can complete this step in one or two commands:

```
chown ketl.ketl /usr/local/ketl
```

Or:

```
chown ketl /usr/local/ketl
```

```
chgrp ketl /usr/local/ketl
```

5. Log off and log back in as this new user.

Installing KETL

To install KETL, complete the following steps:

1. Go to the following location:
http://sourceforge.net/project/showfiles.php?group_id=174441
2. Download the KETL server.
3. Uncompress and unpack the KETL distribution into the new ketl directory:

This step creates the following directories and contents:

Directory	Contents
bin	scripts to control KETL
conf	files to configure the local environment
examples	sample job control files and associated data
lib	Java jar files
license	copies of all product license files
demo	copy of a sample environment
setup	metadata database creation scripts
xml	KETL master data and configuration files

- 4.** Set the root directory of the KETL code tree (KETLDIR).

Shell	File	Syntax
Bourne shell or ksh	.profile	export KETLDIR=/usr/local/KETL
csh or tcsh	.cshrc	setenv KETLDIR /usr/local/KETL

- 5.** Set the default search path to include the KETL binaries and scripts.

Shell	File	Syntax
Bourne shell or ksh	.profile	PATH=\$PATH:\$KETLDIR/bin
csh or tcsh	.cshrc	set path=(\$path \$KETLDIR/bin)

- 6.** Run the source command on the file that contains your updated environment variables.

Shell	Syntax
Bourne shell or ksh	. .profile
csh or tcsh	source .cshrc

Creating the Metadata

You can create the metadata repository in any relational database that supports row-level locking. KETL, however, comes bundled with scripts for both [Oracle](#) and [PostgreSQL](#).

Oracle

To create your initial metadata repository in Oracle, complete the following steps:

1. Log in as a privileged database user.
2. Create an Oracle user “ketlmd.”
This user must be able to connect and create objects (such as table, sequence, and view.)

3. Log in as ketlmd.

4. Go to the following directory:

```
$KETL/setup/Oracle_metadata_repository/
```

5. From a SQL prompt, run the following script:

```
KETL_MD_Schema_Oracle.ddl
```

This script creates and populates the metadata schema.

PostgreSQL

To create your initial metadata repository in PostgreSQL, complete the following steps:

1. Log in as a privileged database user.

2. Go to the following directory:

```
$KETL/setup/PostgreSQL_metadata_repository/
```

3. From a SQL prompt, run the following script:

```
Create_KETL_Owner_PostgreSQL.ddl
```

This script creates the ketlmd user.

4. From a SQL prompt, run the following script:

```
KETL_MD_Schema_PG.ddl
```

This script creates and populates the metadata schema. This script also assigns ownership of the metadata schema to the ketlmd user.

Configuring the Initial Environment

To configure the initial environment, complete the following steps:

1. If KETL does not physically reside on the same machine as the database containing the metadata, then update the following file with the machine name of the KETL server:

```
$KETLDIR/xml/KETLServers.xml
```

Be sure to update the user name and password, as well as the driver and JDBC URL.

2. If you are not using PostgreSQL, then confirm that the following file includes the path of the appropriate thin client driver:

```
$KETLDIR/conf/Extra.Libraries
```

3. Go the following directory:

```
$KETLDIR/bin/
```

4. Confirm that you have execute privileges on all files. Otherwise, run the following command:

```
chmod +x *
```

5. To start the KETL server, run the following command:

```
ketl_start
```

6. Go to the following log file, and check for any errors:

```
$KETLDIR/log/KETL.log
```

Testing the Environment

KETL comes bundled with fully-functional sample files in the following directory:

```
$KETLDIR/examples/xml
```

Of these sample files, the FileWriter.xml job validates a successful installation of the KETL framework, specifically completing the following tasks:

- Read from a small, comma-delimited file (\$KETLDIR/examples/data/CSV.txt).
- Reformat and output three of the five columns to a new file (\$KETLDIR/log/result.txt).
- Output log file directly to the screen.

This section provides sample screenshots after each command in the KETL console.

KETL INSTALLATION GUIDE 2.1

To run this sample job, complete the following steps:

1. Confirm that your environment variables have been set.
2. Confirm that a copy of tools.jar is in the following directory:

```
$KETLDIR/lib
```

3. From the command line, connect to the KETL Console:

```
$ ketl_ctl
```

```
[ketl@localhost log]$ ketl_start
Starting...
nohup: appending output to `nohup.out'
Started.
[ketl@localhost log]$ ketl_ctl
[INFO]Tue May 15 16:52:35 EDT 2007 - [Thread[KETL Console,5,main]] KETL Version
2.1.12 Production, ©2007 Kinetic Networks Inc.
[INFO]Tue May 15 16:52:35 EDT 2007 - [Thread[KETL Console,5,main]] Cache engine
com.kni.etl.ketl.lookup.SleepycatIndexedMap
[INFO]Tue May 15 16:52:35 EDT 2007 - [Thread[KETL Console,5,main]] Plugin com.kn
i.etl.ketl.transformation.geoip.GeoIPTransformation enabled.
KETL Console

->_
```

4. To connect to the default KETL server, enter the following command:

```
-> connect
```

```
[ketl@localhost log]$ ketl_ctl
[INFO]Tue May 15 16:22:10 EDT 2007 - [Thread[KETL Console,5,main]] KETL Version
2.1.12 Production, ©2007 Kinetic Networks Inc.
[INFO]Tue May 15 16:22:10 EDT 2007 - [Thread[KETL Console,5,main]] Cache engine
com.kni.etl.ketl.lookup.SleepycatIndexedMap
[INFO]Tue May 15 16:22:10 EDT 2007 - [Thread[KETL Console,5,main]] Plugin com.kn
i.etl.ketl.transformation.geoip.GeoIPTransformation enabled.
KETL Console

->connect
Connected to localhostPG
->_
```

KETL INSTALLATION GUIDE 2.1

- To check status, enter the following command:

```
-> status
```

```
[INFO]Tue May 15 16:22:10 EDT 2007 - [Thread[KETL Console,5,main]] Cache engine
com.kni.etl.ketl.lookup.SleepycatIndexedMap
[INFO]Tue May 15 16:22:10 EDT 2007 - [Thread[KETL Console,5,main]] Plugin com.kn
i.etl.ketl.transformation.geoip.GeoIPTransformation enabled.
KETL Console

->connect
Connected to localhostPG
->status
KETL Cluster Status
  Registered Servers: 1
  Alive Servers      : 1
  Pending Jobs

  Server      : localhost
  Status      : Active
  Start Time  : 2007-05-15 16:19:20.964
  Last Ping   : 2007-05-15 16:25:11.582553
  Executors (Stats)
    SQL: (Total: 2)
    KETL: (Total: 2)
    XMLSESSIONIZER: (Total: 1)
    OSJOB: (Total: 2)

->_
```

- To initiate the sample job, enter the following command:

```
-> run examples/xml/FileWriter.xml
```

```
[INFO]Tue May 15 16:59:51 EDT 2007 - [Thread[KETL Console,5,main]] Threads initi
alized
[INFO]Tue May 15 16:59:51 EDT 2007 - [FileWrite(0)-Status(0)] Reading file exam
ples/data/CSV.txt
[INFO]Tue May 15 16:59:51 EDT 2007 - [FileWrite(0)-FileWriter] Final Throughput S
tatistics(Records Per Second)
  Overall Read: 8
  Average Read: 8
  Total Records Read: 8
  Overall Write: 8
  Average Write: 8
  Total Records Written: 8
  Thread Statistics
  -----
  Status, Type:com.kni.etl.ketl.reader.NIOFileReader [1 of 1]: 8, errors:
0, timing: N/A
  Target, Type:com.kni.etl.ketl.writer.FileWriter [1 of 1]: 8, errors: 0,
timing: N/A
[INFO]Tue May 15 16:59:51 EDT 2007 - [Thread[KETL Console,5,main]] Job details l
ogged to log/FileWrite.0.joblog
[INFO]Tue May 15 16:59:51 EDT 2007 - [Thread[KETL Console,5,main]] Total executi
on time: 1.676 seconds
[INFO]Tue May 15 16:59:52 EDT 2007 - [Thread[KETL Console,5,main]] Job complete.

->_
```

- Check this log file for any errors.
- Confirm that you have the following file with three columns of text:

```
$KETLDIR/log/result.txt
```

Troubleshooting

This section describes common installation problems:

- [Connectivity Issues](#)
- [Failure to Start the KETL Server](#)
- [Failure to Stop KETL Server](#)
- [Failure to Create Log Files](#)
- [Failure to Create Metadata](#)
- [Missing Jar Files](#)

When you receive errors, be sure to check the log files. The log files are located in the \$KETLDIR/log directory. You can either view the KETL.log file or the KETL.extra.log file (after you have started the KETL server).

Connectivity Issues

If you cannot connect to your metadata repository, you may have one of the following issues:

- [Missing database drivers.](#)
- [Missing path.](#)
- [Invalid connect string.](#)

Missing Database Drivers

To check for a missing driver, complete the following procedure:

1. Go to the directory that contains the thin client JDBC driver.
2. Confirm that you have the appropriate thin client JDBC driver:

Database	Driver
PostgreSQL	postgresql.jar
Oracle	ojdbc14.jar
MySQL	mysql-connector-java-5.0.x-bin.jar

3. If the driver is incorrect or missing, then download the appropriate jar file for your database. Refer to the [Database Drivers](#) section for more information.

Missing Path

If you saved your database driver to the \$KETLDIR/lib directory, then KETL automatically found these files. If you saved your file to a different directory, then KETL may not be able to find it if the environment variable has not been set. To check for a missing path, complete the following procedure.

1. Go to the following file:
`$KETLDIR/conf/Extra.Libraries`
2. Confirm that this file provides the correct path of your jar file. Otherwise, update the path, and save the new changes.
3. Run the appropriate source command on this file.

Invalid Connect String

If your metadata repository does not reside on the same machine as the KETL server, then KETL may not be able to connect to your database. To check your connection, complete the following procedure:

1. Go to the following file:

`$KETLDIR/xml/KETLServers.xml`

2. Confirm that the user name and password have been set correctly.

If, for example, you named the `ketlmd` user to something else, then confirm that you entered the alternate user name and password.

3. Confirm that the driver and JDBC URL have been set correctly.
4. Confirm that your passphrase has been set correctly.
5. Save any new changes.

Failure to Start the KETL Server

If you cannot start KETL, you may have one of the following issues:

- [Disconnected KETL server.](#)
- [Incorrect environment settings.](#)
- [Incorrect permissions.](#)

Disconnected KETL Server

To confirm that you are connected to the KETL server, complete the following procedure:

1. Enter the following command:

```
$ ps -f | grep ETL
```

This command searches all processes and identifies the ETL daemon.

2. Confirm that the ETL daemon is running.

```
[ketl@localhost log]$ ps -f | grep ETL
ketl      3102      1 14 16:35 tty1      00:00:12 /usr/java/jdk1.5.0_10/bin/java -
server -XX:+UseParallelGC -Xms128m -Xmx1024m -Djava.awt.headless=true -Dlog4j.co
nfiguration=file:/home/ketl/ketl_dir/KETLBuild/conf/KETL.log.properties -classpa
th /home/ketl/ketl_dir/KETLBuild/lib/axis-ant.jar:/home/ketl/ketl_dir/KETLBuild/
lib/axis.jar:/home/ketl/ketl_dir/KETLBuild/lib/commons-discovery-0.2.jar:/home/k
etl/ketl_dir/KETLBuild/lib/commons-logging-1.0.4.jar:/home/ketl/ketl_dir/KETLBuild
lib/hsqldb.jar:/home/ketl/ketl_dir/KETLBuild/lib/jaxrpc.jar:/home/ketl/ketl_d
ir/KETLBuild/lib/je-3.2.13.jar:/home/ketl/ketl_dir/KETLBuild/lib/junit.jar:/home
/ketl/ketl_dir/KETLBuild/lib/KETLGeoIP.jar:/home/ketl/ketl_dir/KETLBuild/lib/KET
LGPLExtensions.jar:/home/ketl/ketl_dir/KETLBuild/lib/KETL.jar:/home/ketl/ketl_d
ir/KETLBuild/lib/KETLOracleExt.jar:/home/ketl/ketl_dir/KETLBuild/lib/KETLPostGreE
xt.jar:/home/ketl/ketl_dir/KETLBuild/lib/KETLSOAPExt.jar:/home/ketl/ketl_dir/KET
LBuild/lib/KETLXMLExt.jar:/home/ketl/ketl_dir/KETLBuild/lib/log4j-1.2.9.jar:/hom
e/ketl/ketl_dir/KETLBuild/lib/postgresql.jar:/home/ketl/ketl_dir/KETLBuild/lib/s
aa.jar:/home/ketl/ketl_dir/KETLBuild/lib/saxon8-dom.jar:/home/ketl/ketl_dir/KET
LBuild/lib/saxon8.jar:/home/ketl/ketl_dir/KETLBuild/lib/saxon8-xpath.jar:/home/k
etl/ketl_dir/KETLBuild/lib/tools.jar:/home/ketl/ketl_dir/KETLBuild/lib/wsd14j-1.
5.1.jar:/usr/java/jdk1.5.0_10/lib/tools.jar ETLDaemon
ketl      3137    2516  0 16:37 tty1      00:00:00 grep ETL
[ketl@localhost log]$ _
```

If you cannot locate the ETL daemon, then the KETL server has been disconnected.

Incorrect Environment Settings

If the system cannot run an executable file (the `ketl_ctl` tool, for example), then you may not have the correct environment settings. Typically, you receive the “File not found” error, meaning that the system cannot run an executable file because it cannot locate it.

To check your environment settings, complete the following procedure:

1. Confirm that the `$KETLDIR/bin` environment variable has been set and added to the path.

```
echo $PATH
```

2. Confirm that Java is available.

```
java -version
```

If successful, the version details display.

3. Confirm that `JAVA_HOME` has been set and added to the path. If it has not been set, then `ketl_ctl` cannot find the Java compiler.

Incorrect Permissions

If your environment settings are correct, but you still cannot run the executable file, you may not have the correct permissions. In this case, you receive the “Permission denied” error. All files in the `$KETLDIR/bin` directory must have execute privileges. Confirm that all files have the correct permissions. Otherwise, use the following command to change permissions on all files:

```
chmod +x $KETLDIR/bin/*
```

Failure to Stop the KETL Server

When you try to stop the KETL server, you may find that it hangs. If so, you may have one of the following issues:

- [Incorrect server ID.](#)
- [Invalid privileges.](#)

Incorrect server ID

To stop the KETL server, you need to specify the correct server ID. For example, at the KETL console, to stop server ID 0, enter the following command:

```
-> shutdown 0
```

Invalid Privileges

To stop the KETL server, you must be the same user that started it. Any other user who attempts to stop the KETL server does not have the correct privileges. To confirm that you are the correct user, you can search all processes to identify the user that started the ETL daemon. Refer to the [Disconnected KETL Server](#) section for more information.

Failure to Create Log Files

If the system cannot create log files, the log directory may not have the correct permissions set. You need write permission on the log directory.

Complete the following procedure:

1. Go to the following directory:

```
$KETLDIR/log
```

2. Confirm that the log directory has write privilege. Otherwise, use the following command to change permission:

```
chmod +w log
```

Failure to Create Metadata

If you cannot create your initial metadata in a PostgreSQL database, you may have one of the following issues:

- [Missing procedural language](#).
- [PostgreSQL bug](#).

Missing Procedural Language

After you run the first PostgreSQL script to create the ketlmd user, you may receive an error message that “function moddate() does not exist.” If so, your system does not have the appropriate procedural language installed for the database. Refer to the [Procedural Language](#) section for more information.

```
CREATE ROLE
```

```
psql:Create_KETL_Owner_Postgres.ddl:4: NOTICE: schema "ketlmd" does not exist
```

```
ALTER ROLE
```

```
psql:/home/ketl/ketl_dir/KETLBuild/setup/PostgreSQL_metadata_repository/KETL_MD_Schema_PG.ddl:1212: ERROR: language "plpgsql" does not exist
```

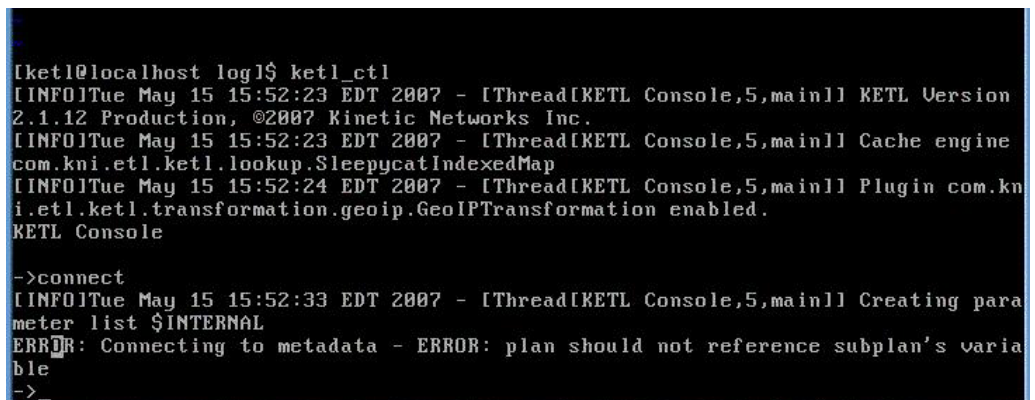
```
HINT: Use CREATE LANGUAGE to load the language into the database.
```

KETL INSTALLATION GUIDE 2.1

```
psql:/home/ketl/ketl_dir/KETLBuild/setup/PostgreSQL_metadata_repository/KE
TL_MD_Schema_PG.ddl:1214: ERROR: function moddate() does not exist
psql:/home/ketl/ketl_dir/KETLBuild/setup/PostgreSQL_metadata_repository/KE
TL_MD_Schema_PG.ddl:1220: ERROR: function moddate() does not exist
psql:/home/ketl/ketl_dir/KETLBuild/setup/PostgreSQL_metadata_repository/KE
TL_MD_Schema_PG.ddl:1226: ERROR: function moddate() does not exist
psql:/home/ketl/ketl_dir/KETLBuild/setup/PostgreSQL_metadata_repository/KE
TL_MD_Schema_PG.ddl:1232: ERROR: function moddate() does not exist
```

PostgreSQL Bug

PostgreSQL has a known bug that requires additional steps before you can create the metadata repository. This bug only happens on rare occasions, such as when working on a clean environment that has not had a database installed. Refer to the following screenshot:



```
[ketl@localhost log]$ ketl_ctl
[INFO]Tue May 15 15:52:23 EDT 2007 - [Thread[KETL Console,5,main]] KETL Version
2.1.12 Production, ©2007 Kinetic Networks Inc.
[INFO]Tue May 15 15:52:23 EDT 2007 - [Thread[KETL Console,5,main]] Cache engine
com.kni.etl.ketl.lookup.SleepycatIndexedMap
[INFO]Tue May 15 15:52:24 EDT 2007 - [Thread[KETL Console,5,main]] Plugin com.kn
i.etl.ketl.transformation.geoip.GeoIPTransformation enabled.
KETL Console

->connect
[INFO]Tue May 15 15:52:33 EDT 2007 - [Thread[KETL Console,5,main]] Creating para
meter list $INTERNAL
ERROR: Connecting to metadata - ERROR: plan should not reference subplan's varia
ble
->
```

After you run the first script to create the user, complete the following steps:

1. Exit the KETL console.
2. At the SQL prompt, insert values into the `ketlmd.parameter_list` table:

```
Insert into ketlmd.parameter_list
      (parameter_list_id, parameter_list_name)
      Values (1, '1234');
```

3. At the SQL prompt, analyze the table:
4. At the SQL prompt, delete the new row:

```
Delete from ketlmd.parameter_list
      where parameter_list_id=1;
```

At this point, you have fixed the bug in PostgreSQL.

5. Start the KETL console:

```
$ ketl_ctl
```

Missing Jar Files

KETL automatically finds all files in the \$KETLDIR/lib directory. If you download any new jar files, be sure copy them into this directory. Otherwise, if you copy the files into other directories, be sure to update the \$KETLDIR/conf/Extra.Libraries file with the correct paths. Refer to the [Missing Path](#) section for more information.

tools.jar

The tools.jar file is required and is part of your Java installation. If you receive an error that tools.jar is not available, then your JAVA_HOME environment variable may not be set. Go to the appropriate file to confirm that your environment variable has been set. If needed, set JAVA_HOME to the correct path and source the file. Refer to the [Complex Transformations](#) section for more information.