

# DBC Crane User's Guide (Version 2.0.0)

---

**Bronzeage L.L.C.**

**<http://www.bronzeage.us>**

**Email: [info@bronzeage.us](mailto:info@bronzeage.us)**

Copyright © 2010 - 2012, Bronzeage L.L.C. All rights reserved.

The specifications and information regarding the products described in this manual are subject to change without notice. All statements, information, and recommendations in this manual are believed to be accurate. Bronzeage makes no representations or warranties of any kind, express or implied, including, without limitation, those of merchantability, fitness for a particular purpose, and non infringement, regarding this manual or the products' use or performance. In no event will Bronzeage be liable for indirect, incidental, consequential, special, or economic damages (including lost business profits, business interruption, loss or damage of data, and the like) arising out of the use or inability to use this manual or the products, regardless of the form of action, whether in contract, tort (including negligence), breach of warranty, or otherwise, even if Bronzeage has been advised of the possibility of such damages.

# Table of Contents

<b>1. INTRODUCTION .....</b>	<b>3</b>
1.1. OVERVIEW .....	3
1.2. PURPOSE .....	3
<b>2. INSTALLING DBCRANE .....</b>	<b>4</b>
2.1. SYSTEM REQUIREMENTS .....	4
2.2. JAVA 6 .....	4
2.3. ON WINDOWS SYSTEM PLATFORM .....	4
2.4. ON UNIX / LINUX PLATFORM .....	4
2.5. JDBC DRIVERS .....	5
<b>3. DBCRANE DIRECTORIES AND FILES STRUCTURE .....</b>	<b>6</b>
<b>4. CREATING OR EDITING DBCRANE CONFIGURATION FILE .....</b>	<b>7</b>
4.1. START DBCRANE GUI .....	7
4.2. ADD CONNECTIONS TO SOURCE AND TARGET DATABASES .....	7
4.3. SAVING AND LOADING CONFIGURATION FILE .....	9
4.4. CREATE REFRESH GROUPS .....	10
4.5. EXTRACT TABLES TO FLAT FILES .....	12
4.6. ADD TABLES TO REFRESH GROUP .....	14
4.7. MODIFY TABLE REPLICATION DETAIL .....	15
4.8. CHANGE TABLE REPLICATION ORDER .....	17
<b>5. RUN REPLICATION .....</b>	<b>18</b>
5.1. RUN REPLICATION IN GUI .....	18
5.2. RUN DBCRANE IN COMMAND LINE .....	18
<b>6. TROUBLESHOOTING .....</b>	<b>21</b>
<b>7. EXAMPLES OF CALLING STORED PROCEDURES .....</b>	<b>22</b>
7.1. EXTRACT DATA FROM STORED PROCEDURE (ORACLE) .....	22
7.2. EXTRACT DATA FROM STORED PROCEDURE (SQL SERVER) .....	23
<b>8. PERFORMANCE TUNING .....</b>	<b>24</b>
8.1. SETTING "COMMIT" PARAMETER TO FALSE .....	24
8.2. REPLICATE MULTIPLE TABLES IN PARALLEL .....	24
8.3. INCREASING BATCH SIZE .....	24
8.4. INCREASING NUMBER OF THREADS .....	24
8.5. CONFIGURE JVM MEMORY SIZE .....	25
8.6. DEBUGGING MODE .....	25
<b>9. APPENDIX .....</b>	<b>27</b>
APPENDIX A: SPECIAL CHARACTERS .....	27
APPENDIX B: DATE/TIME FORMAT .....	28
APPENDIX C: SOFTWARE LICENSE KEY .....	29
APPENDIX D: VIEWING THE VERSION .....	29

# 1. Introduction

---

## 1.1. Overview

---

DBC Crane is an ultra high speed data replication for large database tables in heterogeneous environments.

- DBC Crane works with any databases and data warehouses that come with JDBC driver, including Oracle, DB2, MSSQL, MySQL, Teradata, Netezza, Sybase, PostgreSQL
- DBC Crane supports any operating systems that come with Java version 1.6, including Solaris, HP-UX, AIX, Tru64, Windows, Linux, Apple Mac
- DBC Crane does direct in-memory streaming into target databases with very high speed and throughput while allowing automatic transformation such as rounding or conversions and at same time completely eliminating the need for intermediate storage. E.g. DBC Crane is at least 3 times faster than Oracle export / import when both source and target databases are Oracle.
- DBC Crane is capable of performing fast parallel table replication which effectively utilizes multi-processor (multi-core) environments.
- DBC Crane stores all configuration information in an XML file, with passwords encrypted, eliminating the need for a repository database.
- DBC Crane doesn't create any tables or other database objects in source or target databases.
- DBC Crane is the fastest ETL tool for RDBMS.
- DBC Crane is easy to set up and use.

## 1.2. Purpose

---

This document was created for Database and System Administrators, Data Warehouse Architects, Developers.

## 2. Installing DBCrane

---

### 2.1. System Requirements

---

- **Software**

JVM (Java Virtual Machine) version 1.6 or higher  
JDBC drivers

- **Hardware**

Memory (RAM) -- 1GB or more. For replication of terabyte-sized tables, 64GB memory is recommended.

### 2.2. Java 6

---

DBC Crane is a Java application and requires Java runtime to be installed on the computer. The minimal JRE required version is 1.6. The executable `java.exe` or `java` need to be included in the `PATH` environment variable.

```
C:\>java -version
java version "1.6.0_24"
Java(TM) SE Runtime Environment (build 1.6.0_24-b07)
Java HotSpot(TM) Client VM (build 19.1-b02, mixed mode, sharing)
```

### 2.3. On Windows System Platform

---

DBC Crane is shipped as zip file.

Create a new directory DBC Crane and extract all files from the archive into that directory.

### 2.4. On UNIX / LINUX Platform

---

DBC Crane is shipped as a tar archive file.

Create a new directory DBC Crane and extract all files from the archive into that directory.

```
$ mkdir DBCrane
$ cd DBCrane
$ tar xvf DBCrane.tar
```

## 2.5. JDBC drivers

---

DBC Crane is shipped with JDBC drivers for Oracle and SQL Server. For other JDBC drivers, please contact your database vendor for instructions where to download the appropriate drivers. Make sure that the JDBC drivers support Java 6. Place the JDBC drivers in DBC Crane/lib directory.

## 3. DBCrane Directories and Files structure

---

DBC Crane root directory contains subdirectories, executable and other files that the user needs to be aware of.

- **Executable files**

dbcraneUI.sh (or dbcraneUI.cmd in Windows environment) – starts DBC Crane GUI.

dbcrane.sh (or dbcrane.cmd in Windows environment) – starts DBC Crane command line interface.

- **Other files**

DbCrane.jar – the main java library for DBC Crane.

DbCrane.key - contains DBC Crane license key

- **Subdirectories**

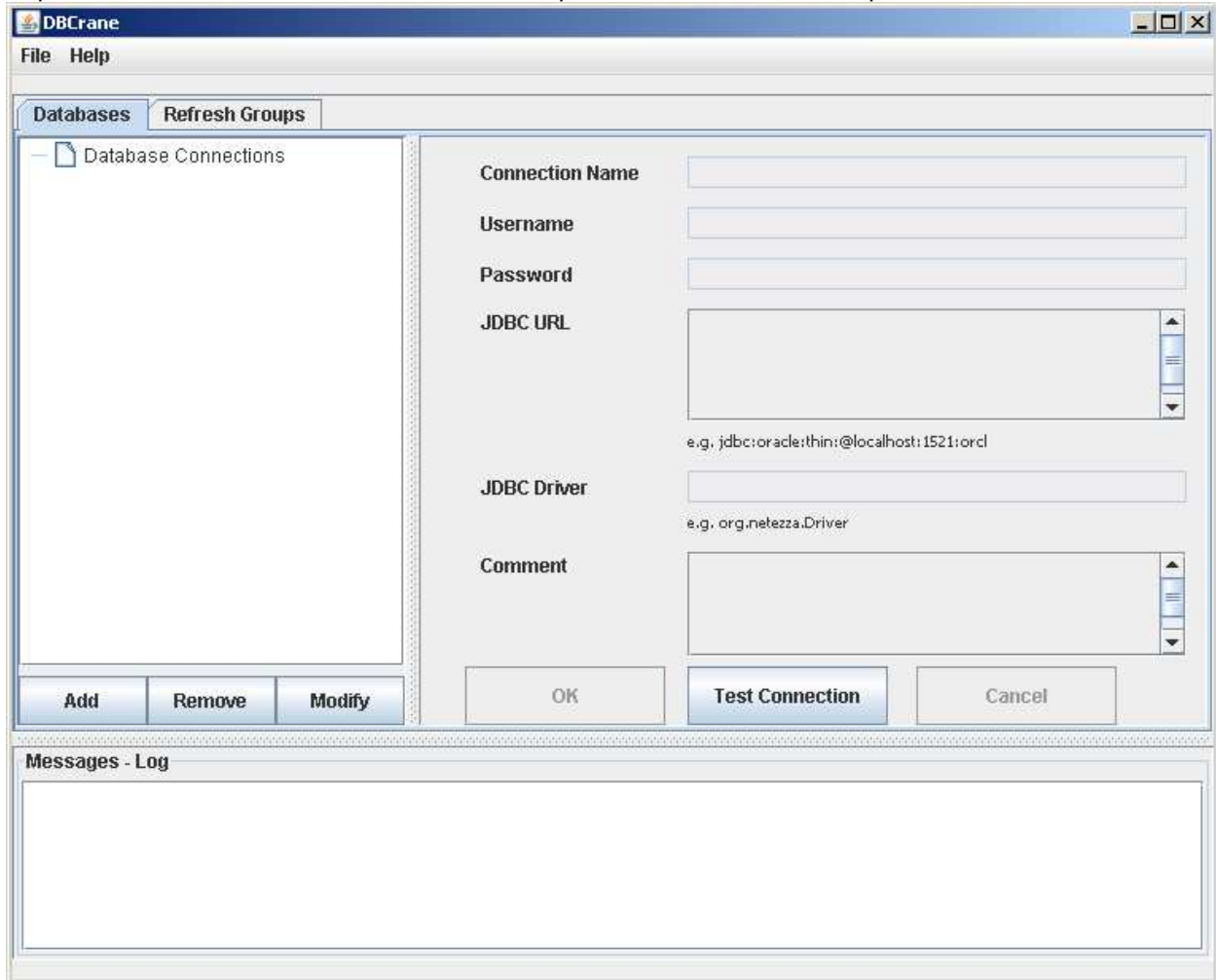
doc – contains DBC Crane documents.

lib – contains java libraries for DBC Crane, including jdbc drivers to connect to databases.

## 4. Creating or editing DBCrane configuration file

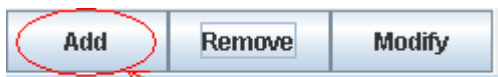
### 4.1. Start DBCrane GUI

Run `dbc craneUI.cmd` in a Windows environment or `dbc craneUI.sh` in a UNIX environment. DBCrane GUI requires Java runtime to be installed on the computer. The minimal JRE required version is 1.6.



### 4.2. Add connections to source and target databases

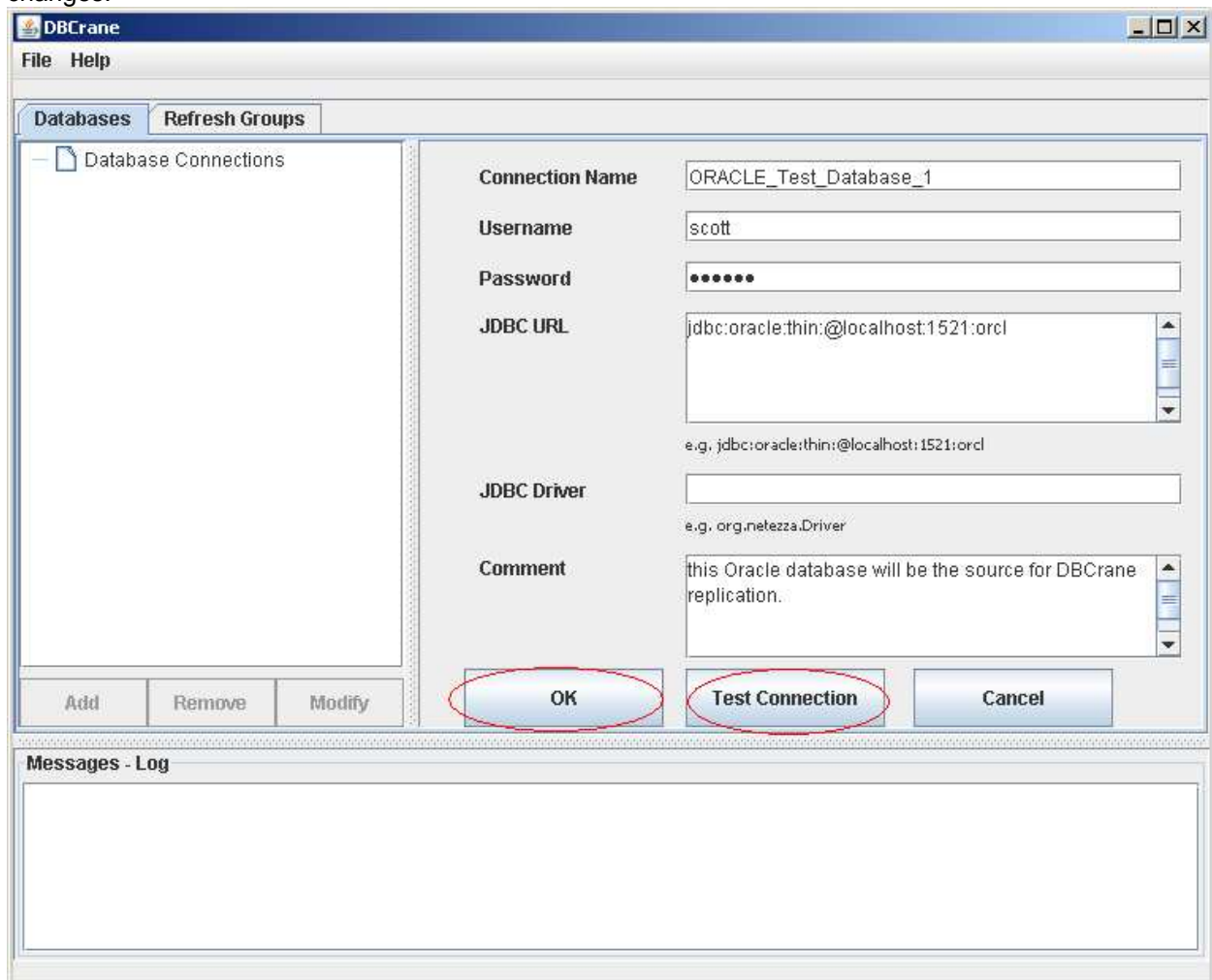
Click "Add" button to create new database connections.



Please fill all the required fields.

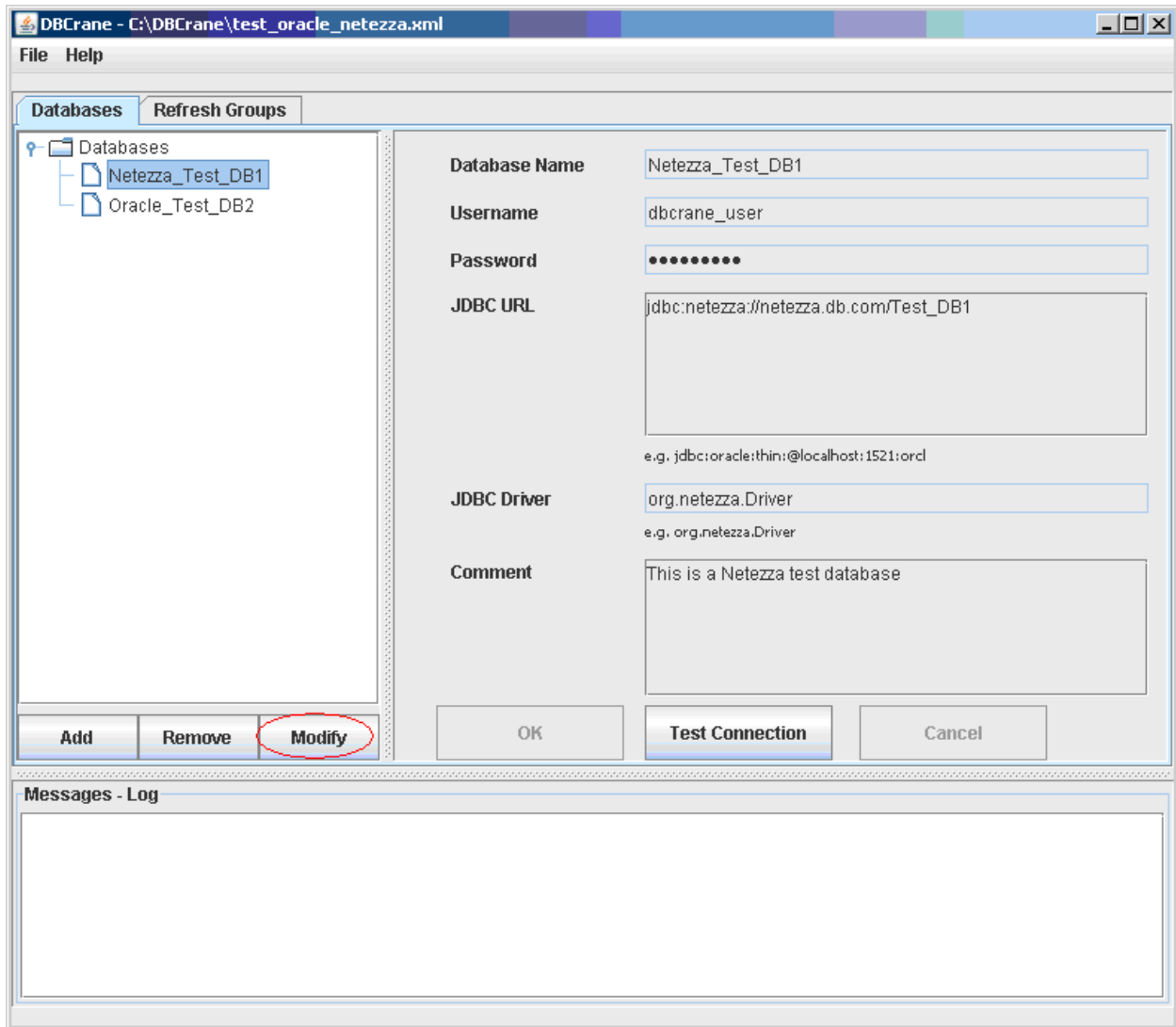
Name	Description	Required
Connection Name	You can give any name to this database connection	YES
Username	Database user name	YES
Password	Password for the database user	YES
JDBC URL	The exact syntax of a JDBC connection URL is specified by your DBMS. For instance, the syntax for Oracle is: jdbc:oracle:thin:@localhost:1521:orcl	YES
JDBC Driver	JDBC drivers prior to 4.0 need to be loaded with Class.forName(...). The exact systax is specified by your DBMS. For instance, the syntax for Netezza is: org.netezza.Driver	NO
Comment	Comment field	NO

Click "Test Connection" button to test the connection to this database. Click "OK" button to confirm the changes.



After "OK" button is clicked, the database detail panel will be grayed out. Click "Modify" button if you need to edit the database connection details.

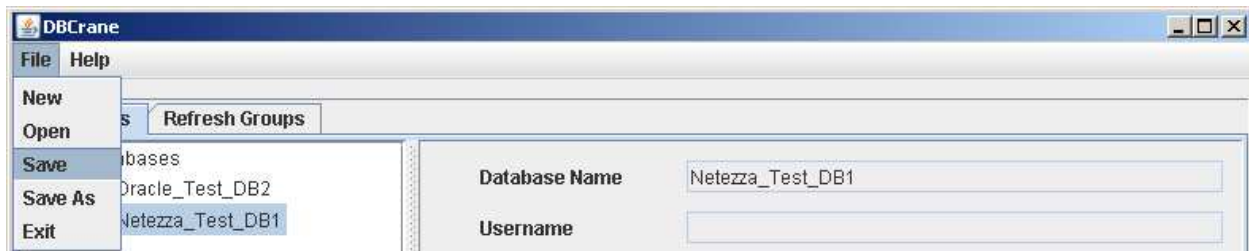




Repeat above steps to add all source and target databases.

### 4.3. Saving and Loading configuration file

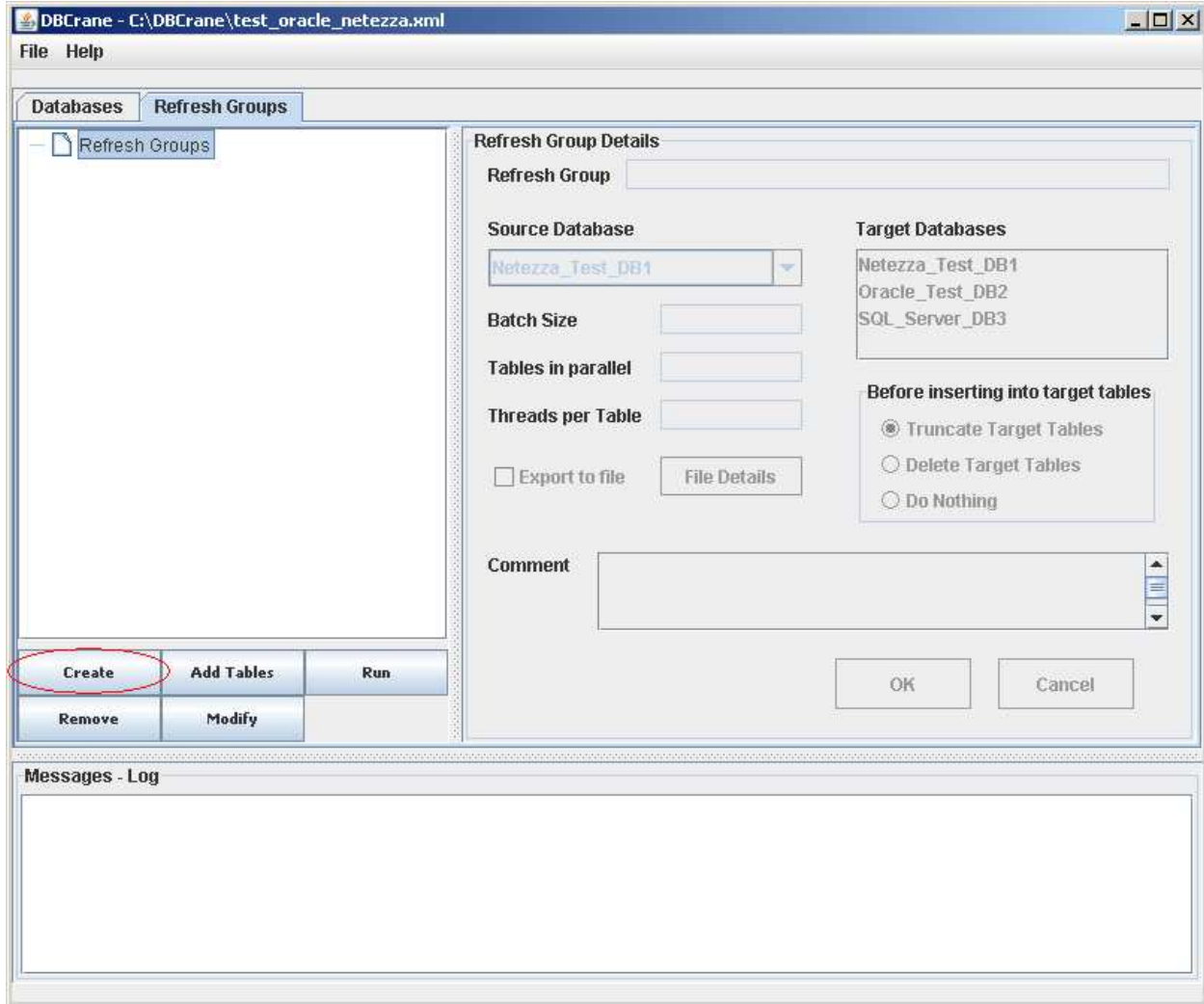
Choose the "Save" or "Save As" option in the "File" menu to save the configuration to an XML file.



Choose the "Open" option in the "File" menu to load a configuration XML file into DBCrane GUI.

## 4.4. Create Refresh Groups

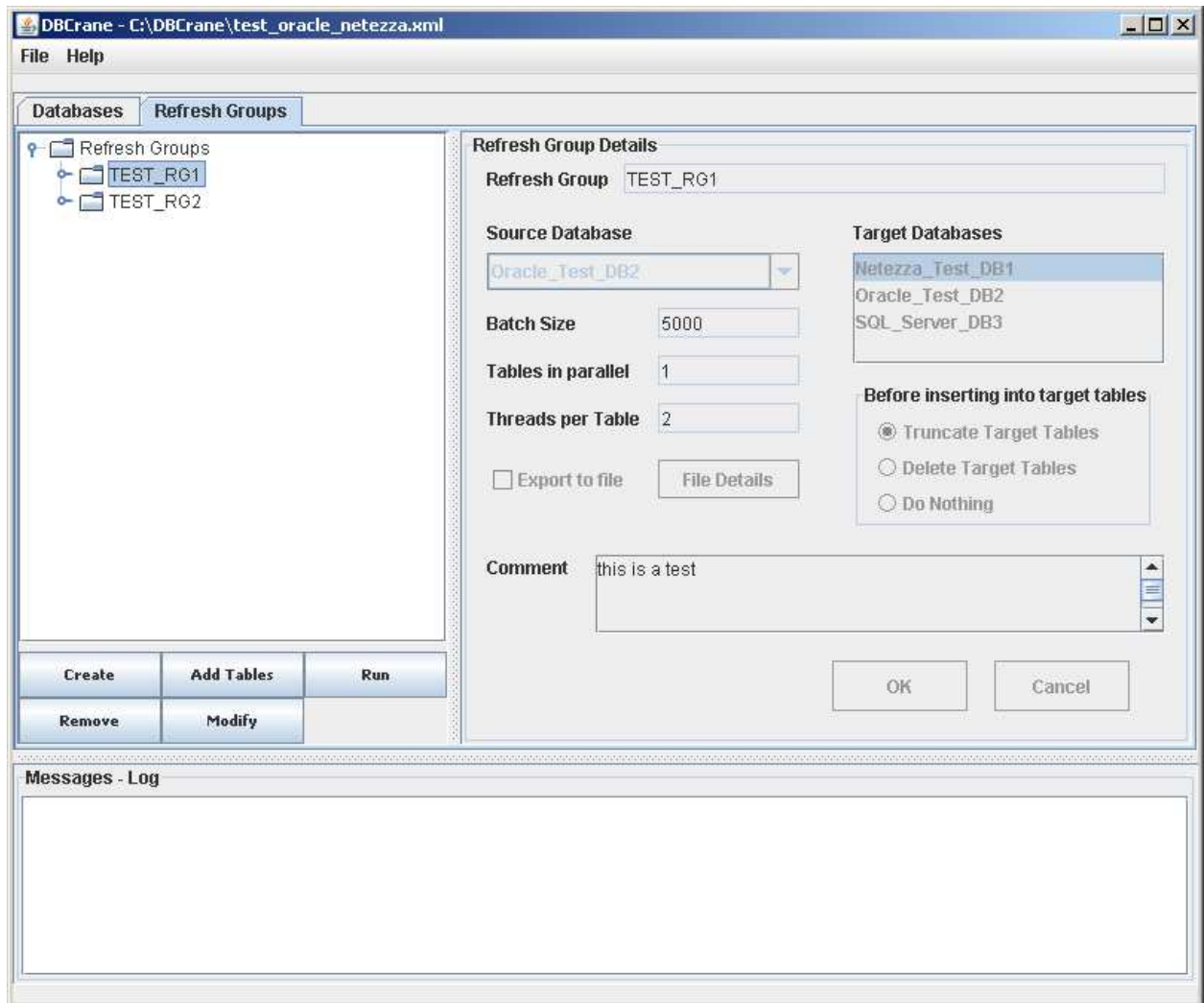
Switch to "Refresh Groups" Tab and Click "Create" button to create new refresh group.



Please fill all the required fields and click "OK" button to confirm the changes. After "OK" button is clicked, the "Refresh Group Details" panel will be grayed out. Click "Modify" button if you need to update the details.

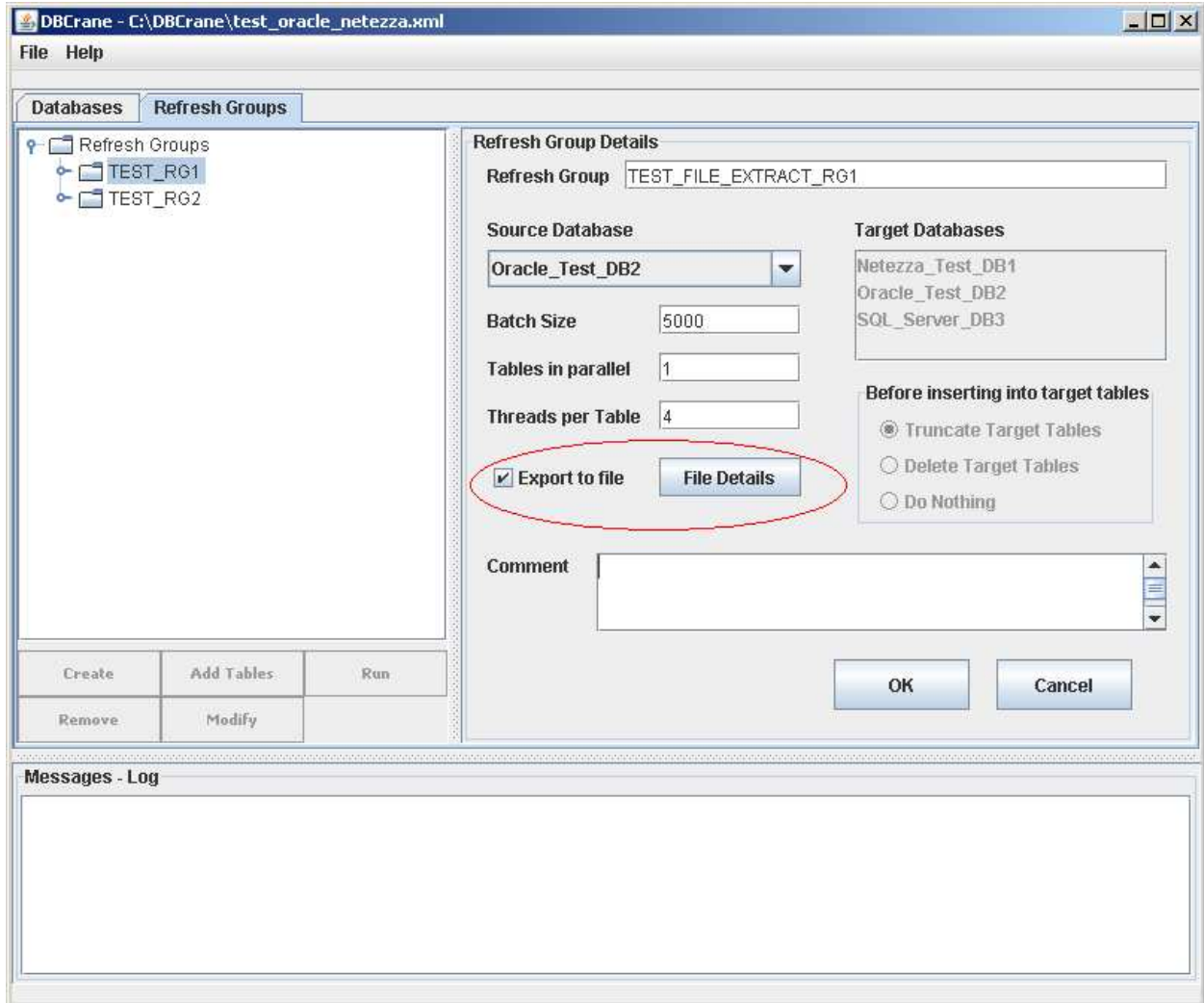
Name	Description	Required
Refresh Group Name	You can give any name to this Refresh Group	YES
Source Database	Choose the source database from the dropdown box.	YES
Target Databases	Select the target databases from the List. You can select more than one target databases.	YES
Batch Size	The number of rows processed in one batch transaction. Performance can be improved by increasing this value, but memory consumption would be increased too and eventually JVM (Java Virtual Machine) may run out of	YES

	memory.	
Tables in parallel	Number of tables replicated in parallel.	YES
Threads per table	Number of threads to replicate a single table. Performance can be improved by increasing this number.	YES
Export to file	Check this check box if tables need to be exported to flat files instead of being replicated to target databases. Click the "File Details" button to specify export file details.	NO
What to do on target tables before inserting data?	There are 3 options: 1. Run a "truncate table" for each table on target databases. 2. Run a "delete table" for each table on target databases. 3. Do nothing.	YES
Comment	Comment field	NO

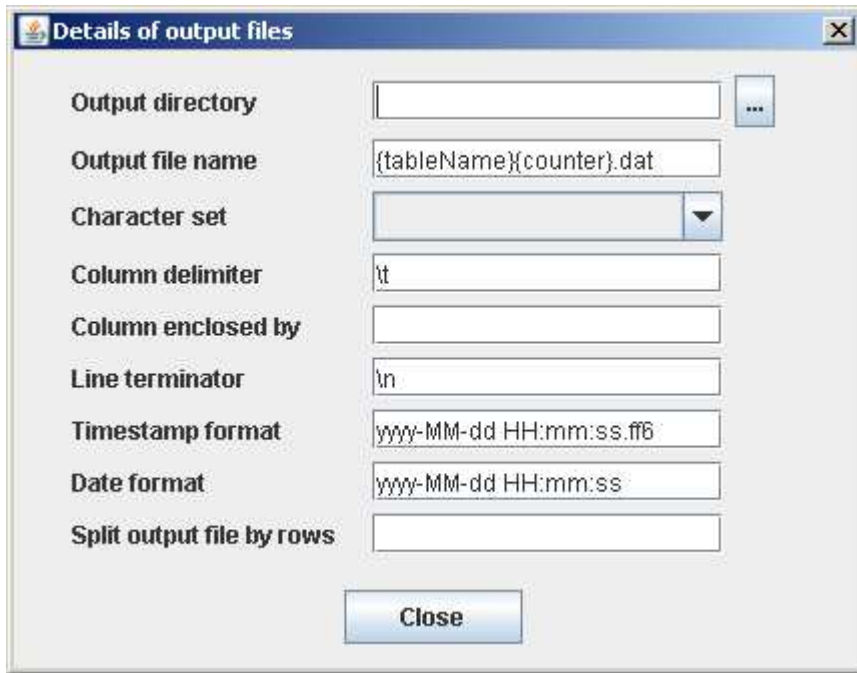


## 4.5. Extract tables to flat files

If tables need to be extracted to flat files, check the check box “Export to file” and click the “File Details” button. The Target Databases section would be grayed out since replication to target databases is not allowed when the “Export to file” option is chosen.



A new window will open after the “File Details” button is clicked. You can specify output file details in this window.

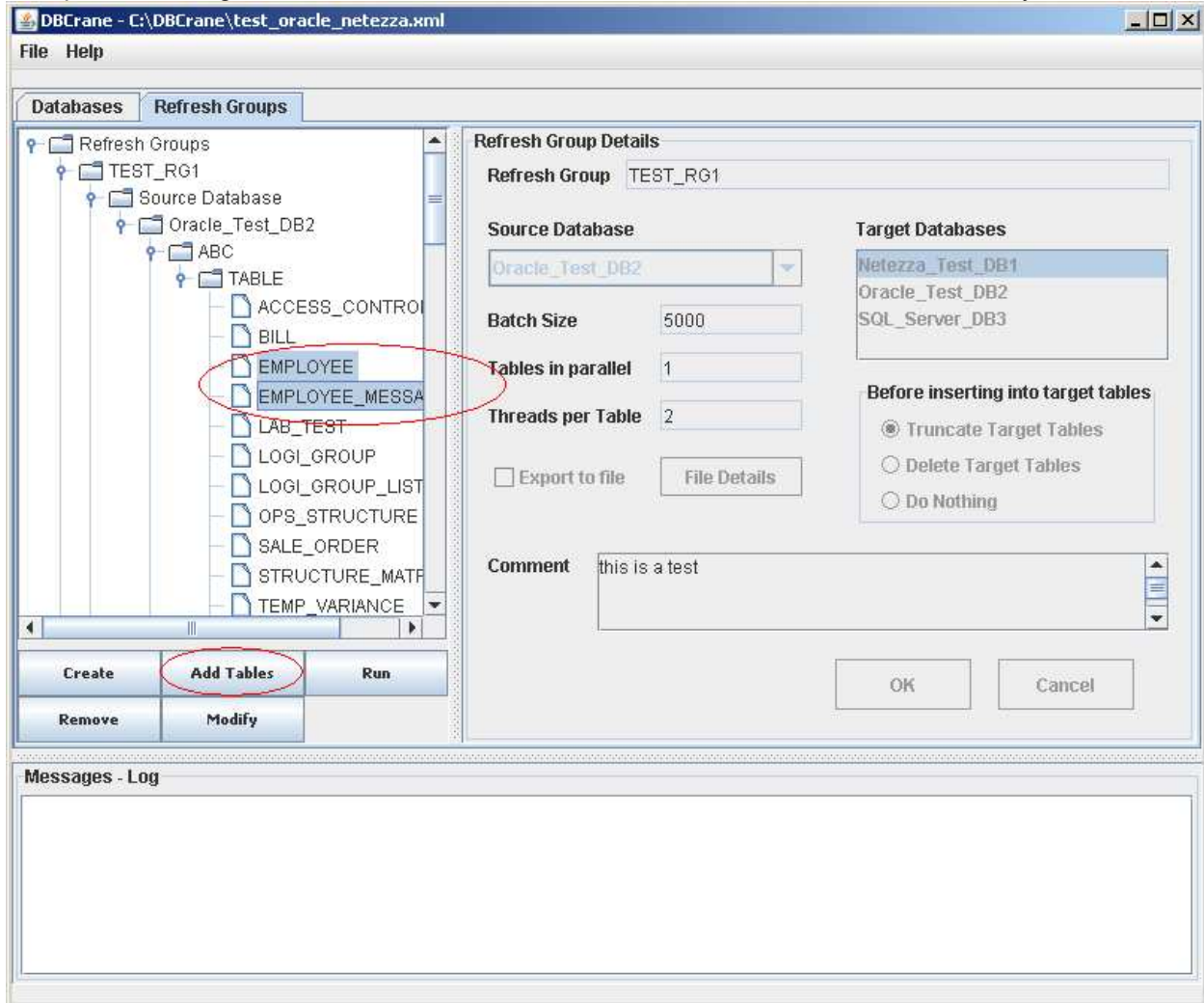


Name	Description	Default
Output directory	All output files will be created in this directory (folder).	The directory where DBCrane is installed
Output file name	Specify names of the output files. <b>{tableName}</b> will be replaced by the table name during extraction. <b>{counter}</b> will be replaced by a sequence number, beginning at 0.	<b>{tableName}{counter}.dat</b>
Character set	Select the character set from the List if you don't want to use the default character set.	Character set of the local machine.
Column delimiter	Specify one or more characters as column delimiter in output files. If special / nonprintable character is required, refer to <a href="#">Appendix A: Special characters</a>	,
Column enclosed by	Specify one or more characters to enclose column in output files. If special / nonprintable character is required, refer to <a href="#">Appendix A: Special characters</a>	Null
Line terminator	Specify one or more characters as end of line in output files. If special / nonprintable character is required, please refer to <a href="#">Appendix A: Special characters</a>	\n ( <i>Enter</i> )
Timestamp format	refer to <a href="#">Appendix B: Date/Time format</a> for more details.	yyyy-MM-dd HH:mm:ss.ff6
Date format	refer to <a href="#">Appendix B: Date/Time format</a> for more details.	yyyy-MM-dd HH:mm:ss
Split output file by rows	Split output file into multiple files instead of a single file. This setting is in number of rows.	Null

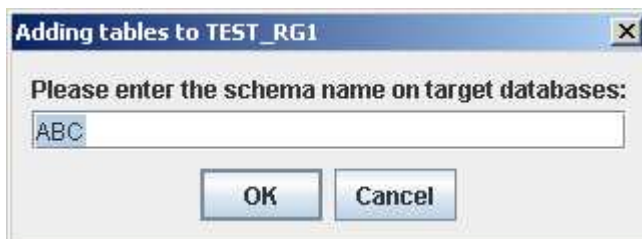
## 4.6. Add tables to Refresh Group

The Refresh Group created in previous step has no tables yet. Tables need to be added to the Refresh Group before running the refresh.

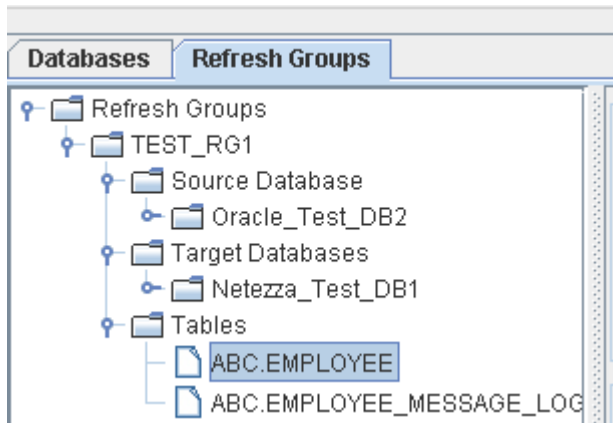
Expand the tree node of “Source Database” and its child nodes. Select/highlight the tables that need to be replicated to target databases. You can select more than one table with Shift or Ctrl key.



Click “Add Tables” button to add the tables to the Refresh Group. This opens a new screen. Enter the schema that these tables should be replicated to and click “OK” button to confirm adding these tables to the Refresh Group.

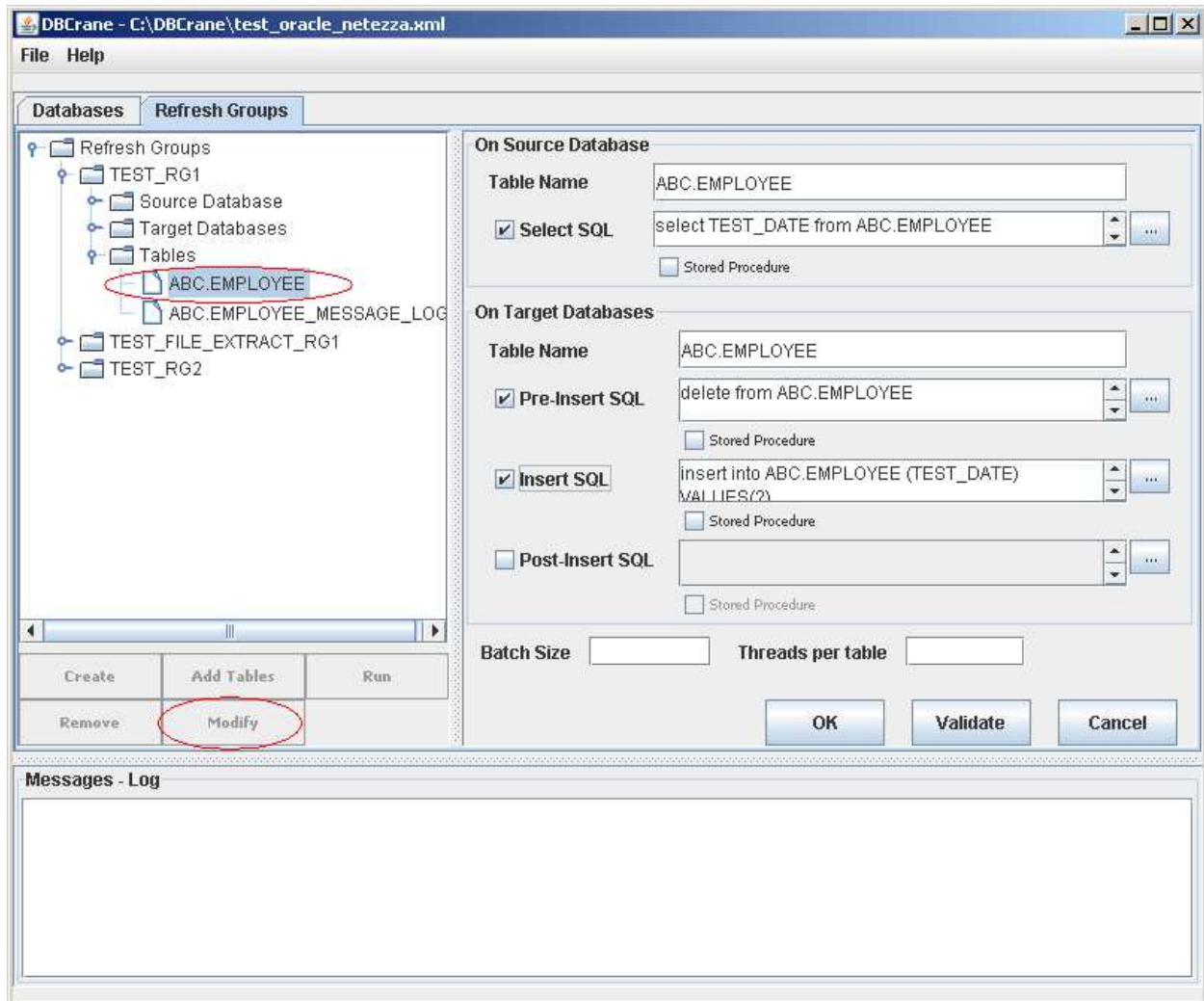


Tables added to the Refresh Group can be seen under the “Tables” tree node.



## 4.7. Modify table replication detail

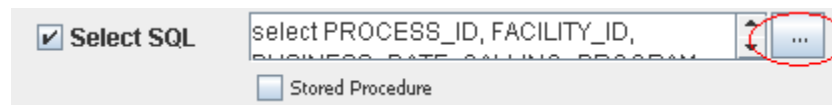
Select/highlight a table on the tree and click “Modify” button to change how this table is replicated to target databases.



Update any of these fields and click "OK" button to confirm the changes. If the field is empty, its default value will be used in replication.

Name	Description	Default Value	Required
Table Name (On Source Database)	The name of the table on source database	N/A	YES
Select SQL (On Source Database)	The SQL to be used to extract data from source database. Check the "Stored Procedure" check box if this is a Stored Procedure. The Stored Procedure will need to return at least one result set. For Oracle database, the stored procedure will need to return a Ref Cursor.	Select <i>&lt;all columns&gt;</i> from <i>&lt;source table name&gt;</i>	NO
Table Name (On Target Databases)	The name of the table on target databases.	The source table name	NO
Pre-Insert SQL (On Target Databases)	The SQL to be executed on target databases before replication begins. Most users use this field to delete data on the target table. Check the "Stored Procedure" check box if this is a Stored Procedure.	Delete from <i>&lt;target table name&gt;</i> or truncate table <i>&lt; target table name&gt;</i> depending on the configuration in "Refresh Group Details"	NO
Insert SQL (On Target Databases)	The SQL statement to be executed on target databases to insert rows. Values are represented by "?". Check the "Stored Procedure" check box if this is a Stored Procedure.	Insert into <i>&lt; target table name&gt;</i> ( <i>&lt;all columns&gt;</i> ) values (?,?,...)	NO
Post-Insert SQL (On Target Databases)	The SQL statement to be executed on target databases after replication completes successfully. Check the "Stored Procedure" check box if this is a Stored Procedure.	N/A	NO
Batch Size	Refer to <i>4.4. Create Refresh Groups</i> for details of this parameter.	From Refresh Group	NO
Threads per table	Refer to <i>4.4. Create Refresh Groups</i> for details of this parameter.	From Refresh Group	NO

You can click edit button to open the edit window.



Click "Validate" button to make sure the changes are valid. If the table does not exist in target databases, the "Validate" process would attempt to create the table.





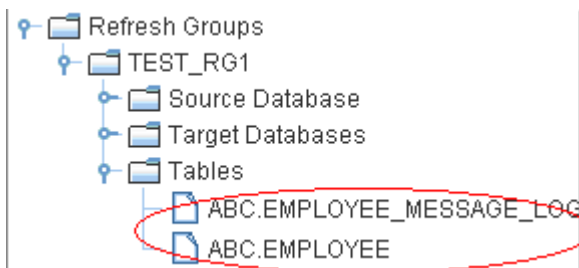
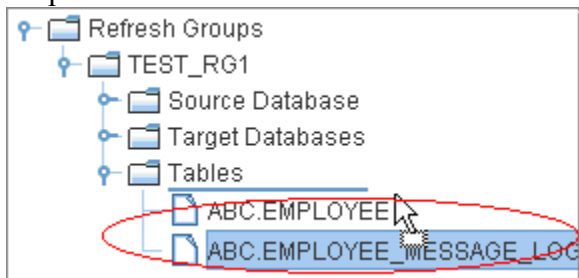
Click “Yes” and the “Create table” window would pop up. The “create table” SQL statement is editable.



Click “OK” to run the create table SQL in target database.

## 4.8. Change table replication order

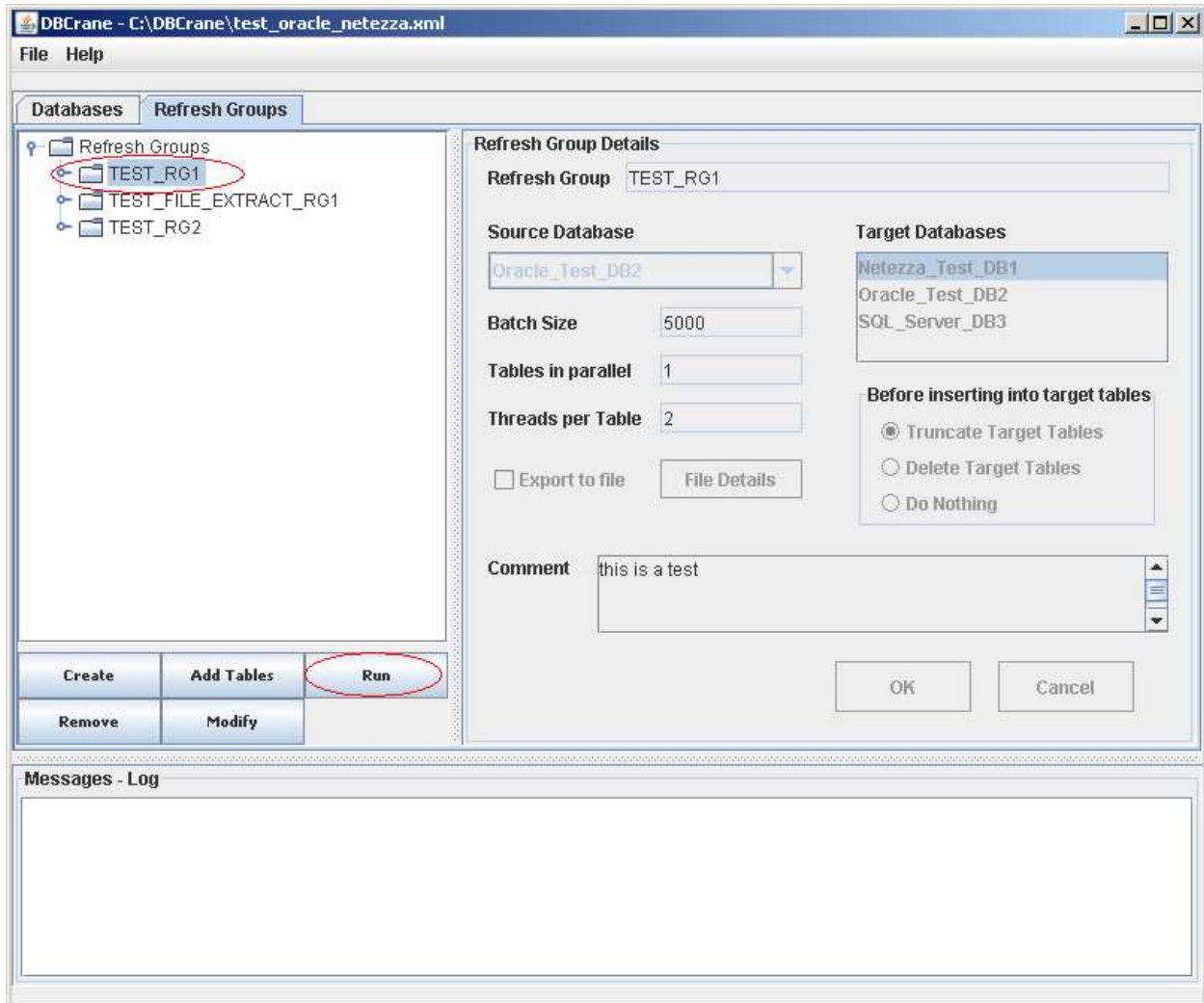
Tables are replicated in the order appeared on the “Tree”. You can change the order by drag-and-drop.



## 5. Run replication

### 5.1. Run replication in GUI

Select a refresh group on the tree and click “Run” button to run it.



### 5.2. Run DBCrane in Command Line

Bronzeage recommends using DBCrane command line for large tables. DBCrane command line can be called by any scheduler tools, including cron, Windows Tasks, Control-M and autosys.

- On Windows platform, run *dbcrane.cmd* with parameters.
- On Unix and Linux operating systems, run *dbcrane.sh* with parameters.

**DBC Crane User's Guide**  
**Version 2.0.0**

- On other systems, run `java -Xms32M -Xmx512M -jar DbCrane.jar` with parameters.

Run DBCrane in command line with `-h` option for usage details.

```
C:\DBC Crane>dbcrane.cmd -h
```

```
DBC Crane
Version 2.0.0
Copyright (c) 2010 - 2012, BronzeAge L.L.C.
For more information, please visit www.bronzeage.us.
This trial license will expire on Mar-27-2012
```

Usage:

```
[-h|--help] --config <configXML> [--parallel <parallel>] [--batchsize <batchSize>]
[--stoponerror <stopOnError>] [--log <logFile>] [--commit <commit>] [--debug <debugMode>]
refreshGroup1 refreshGroup2 ... refreshGroupN
```

```
[-h|--help]
Prints this help message.
```

```
--config <configXML>
Configuration File in XML format.
```

```
[--parallel <parallel>]
The number of parallel threads. (default: 0)
```

```
[--batchsize <batchSize>]
The number of rows processed in one batch transaction. (default: 0)
```

```
[--stoponerror <stopOnError>]
Stop execution after an error occurs. If set to false, continue
replication of the next table. (default: false)
```

```
[--log <logFile>]
Log file name.
```

```
[--commit <commit>]
Commit after each batch insert. If set to false, commit is performed
after loading each table. (default: true)
```

```
[--debug <debugMode>]
Debugging mode. (default: false)
```

```
refreshGroup1 refreshGroup2 ... refreshGroupN
One or more refresh groups.
```

Parameter	Description	Default Value	Required
config	The configuration file in XML format that is created by DBCrane GUI.	N/A	YES
parallel	DBC Crane can replicate multiple tables in parallel. The number of threads used is controlled by this parameter.	1	NO
batchsize	Overwrite the batch size in Refresh Group. When this parameter is unset or set to 0, the batch size from Refresh Group will be used.	0	NO
stoponerror	When set to true, DBC Crane exits whenever an error occurs.	false	NO

log	log file of screen output.	N/A	NO
commit	Specify whether DBCrane should perform a commit after each batch insert on target databases. If set to false, commit is performed after loading entire table on target database. For most database systems, setting this parameter to false would greatly improve performance.	true	NO
debug	Turn on debugging mode	false	NO
RefreshGroup1 , RefreshGroup2 ...	One of more Refresh Groups in the configuration XML file. They will be run in the order listed here.	N/A	YES

## EXAMPLES

1. Run refresh group TEST\_RG1 created in previous step.

```
C:\DBC Crane>dbcrane.cmd --config test_oracle_netezza.xml TEST_RG1
```

2. Run refresh group TEST\_RG1 in debugging mode.

```
C:\DBC Crane>dbcrane.cmd --config test_oracle_netezza.xml --debug true TEST_RG1
```

3. Run refresh group TEST\_RG1 and TEST\_RG2.

```
C:\DBC Crane>dbcrane.cmd --config test_oracle_netezza.xml TEST_RG1,TEST_RG2
```

## 6. Troubleshooting

---

**Issue:** Following error received while starting *DBC Crane*:

*'java' is not recognized as an internal or external command, operable program or batch file.*

**Solution:** *DBC Crane* is a Java application and requires the Java runtime to be installed on the computer. If Java is installed, then most likely the executable `java.exe` or `java` is not included in the PATH environment variable. *DBC Crane* is supported with Java starting from JRE 1.6 and up.

**Issue:** System cannot have X-Windows installed, which doesn't allow running `dbcraneUI.sh` to edit the configuration XML file.

**Solution:**

1. Start `dbcraneUI.sh` or `dbcraneUI.cmd` (on windows) on another box, create or edit the configuration XML file.
2. Copy or ftp the created XML file to the desired machine where *DBC Crane* (command line) will be executed.
3. Start `dbcrane.sh` on the machine with the just copied XML file.

**Issue:** While starting *DBC Crane*, the following error message is received:

*Exception in thread "main" java.lang.ClassFormatError: us.bronzeage.dbcrane.Main (unrecognized class file version).*

**Solution:** Most likely java is in version prior to 1.6. Run "`java -version`" to check Java version. *DBC Crane* is supported with Java starting from JRE 1.6 and up.

**Issue:** System does not have enough memory (RAM) to run *DBC Crane*. The following error is received:

*Invalid maximum heap size: -Xmx512M  
The specified size exceeds the maximum representable size.  
Could not create the Java virtual machine.*

**Solution:** You can either add more memory to the machine or reduce the memory usage by *DBC Crane*. Refer to [8.4. Configure JVM memory size](#) for how to adjust memory usage by *DBC Crane*.

**Issue:** JVM (Java Virtual Machine) runs out of memory, the following error message is received:

*java.util.concurrent.ExecutionException: java.lang.OutOfMemoryError: Java heap space  
Caused by: java.lang.OutOfMemoryError: Java heap space*

**Solution:** Increase JVM memory heap size. Refer to [8.4. Configure JVM memory size](#) for the details.

## 7. Examples of calling stored procedures

### 7.1. Extract data from stored procedure (Oracle)

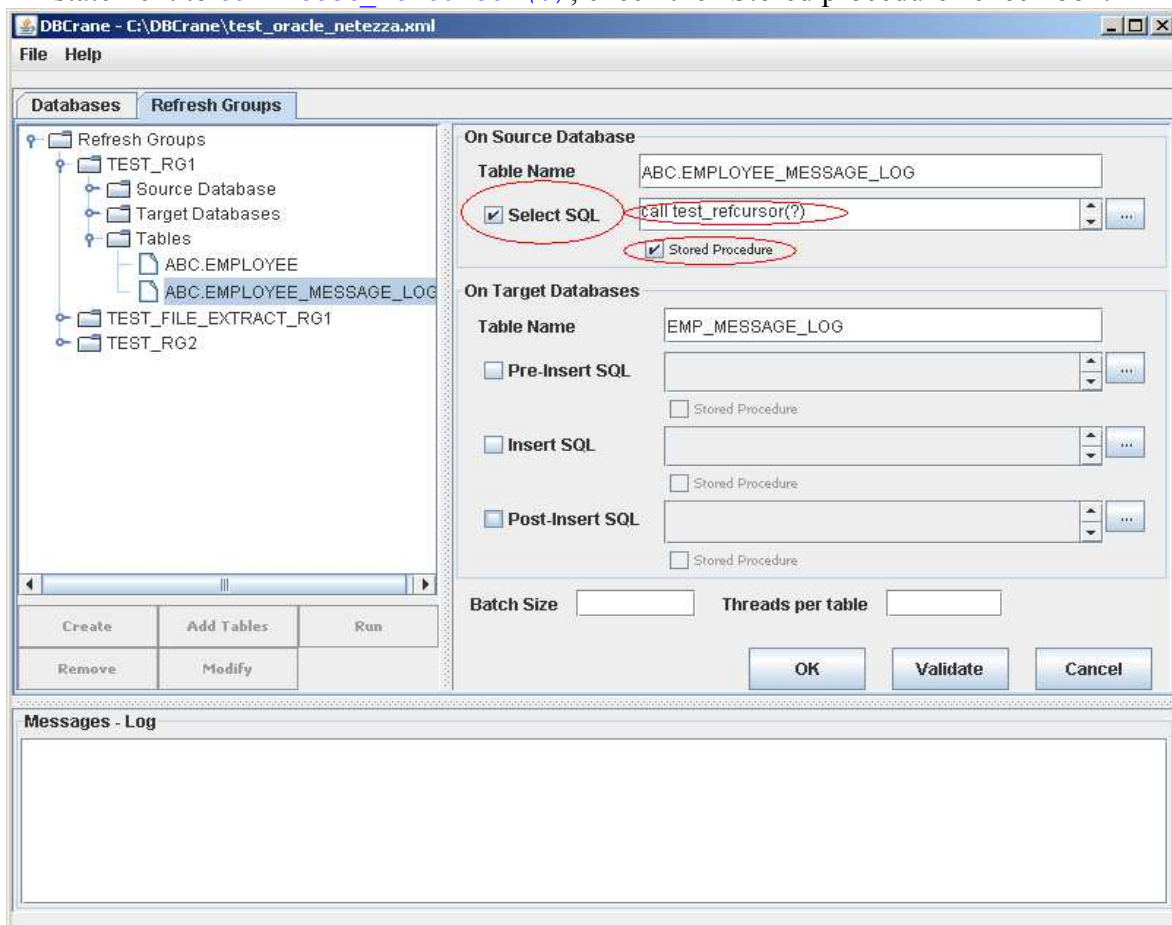
Unlike other database systems, Oracle's stored procedure cannot return result set. This example assumes the following:

- Source database is Oracle.
- You want to extract data from stored procedure instead of table.
- The stored procedure returns a Ref Cursor.

1. Create the stored procedure in the source Oracle database.

```
CREATE OR REPLACE PROCEDURE test_refcursor (p_recordset OUT SYS_REFCURSOR) AS  
BEGIN  
  OPEN p_recordset FOR  
  SELECT *  
  FROM ABC.EMPLOYEE_MESSAGE_LOG;  
END test_refcursor;  
/
```

2. Modify table replication details: Check the "Select SQL" checkbox; change the Select SQL statement to `call test_refcursor(?)`; check the "Stored procedure" check box.



3. Click "Validate" button to make sure the changes are valid and then click "OK" button to confirm the changes.

## 7.2. Extract data from stored procedure (SQL Server)

This example assumes the following:

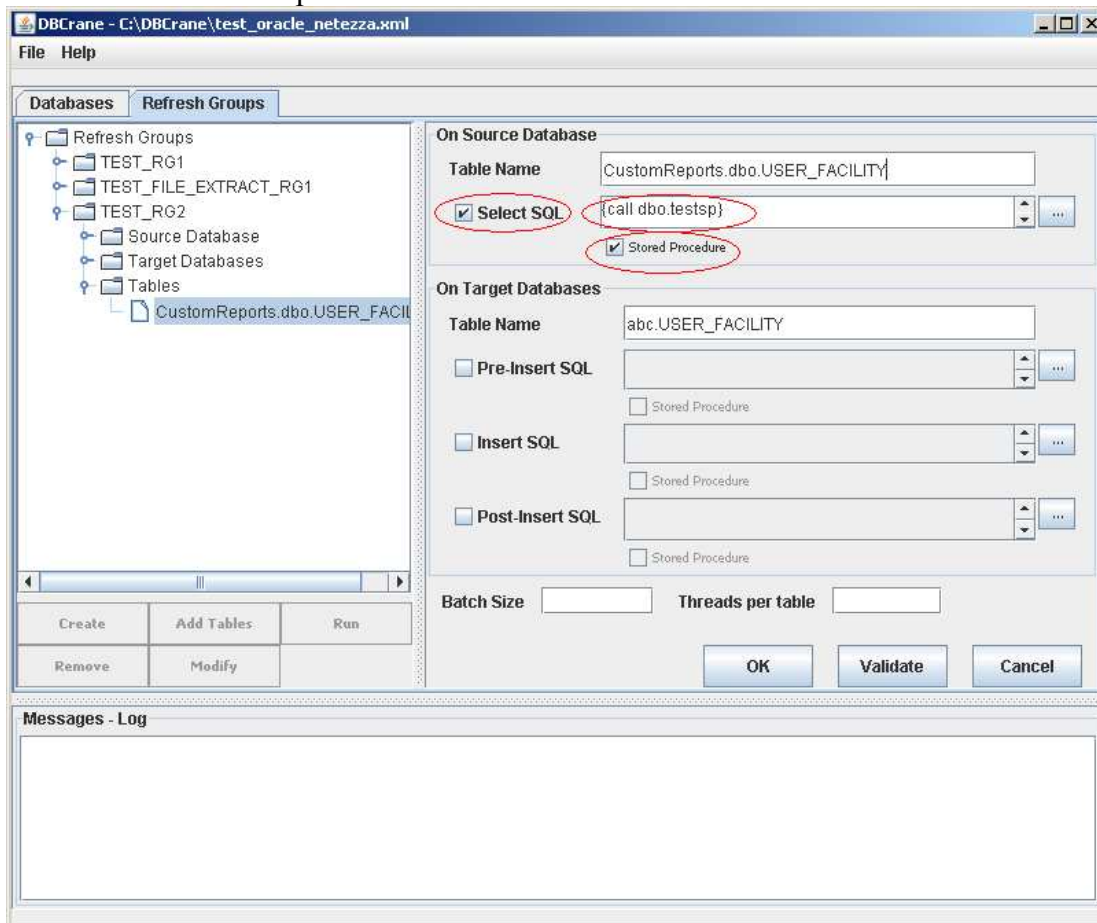
- Source database is SQL Server.
- You want to extract data from stored procedure instead of table.
- The stored procedure returns a result set.

1. Create the stored procedure in the source SQL Server database.

```
CREATE PROCEDURE dbo.testsp
AS
BEGIN
    SELECT *
    FROM dbo.USER_FACILITY
END
```

2. Modify table replication details:

- Check the "Select SQL" checkbox
- Change the Select SQL statement to `{call dbo.testsp}`
- Check the "Stored procedure" check box.



## 8. Performance Tuning

---

This section describes detailed ways to optimize DBCrane replication performance.

### 8.1. Setting “commit” parameter to false

---

By default, commit is performed after each batch insert and may be performed many times for a large table. E.g. if batch size is set to 1,000 and a table has 20 million rows to be replicated, commit would be performed 20,000 (20,000,000 / 1,000) times. For most database systems, commit is a time consuming task and lots of commits would degrade performance significantly. If “commit” parameter is set to false, DBCrane commits only after loading entire table and could improve replication performance by 50% for large tables.

For instance, to set “commit” parameter to false:

```
C:\DBC Crane>dbcrane.cmd --config test_oracle_netezza.xml --commit false TEST_RG1
```

Note: you cannot set “commit” parameter to false in DBCrane GUI.

### 8.2. Replicate multiple tables in parallel

---

DBC Crane can replicate multiple tables in parallel. Refer to [4.4. Create Refresh Groups](#) for details of parameter “Tables in parallel”.

You can also change this parameter in command line. For instance, to replicate 4 tables in parallel:

```
C:\DBC Crane>dbcrane.cmd --config test_oracle_netezza.xml --parallel 4 TEST_RG1
```

### 8.3. Increasing batch size

---

Batch size determines the number of rows replicated in one batch transaction. Increasing batch size would reduce network round trips and other overheads greatly. By increasing batch size from 500 to 100,000, performance would be improved by 5 to 10 times.

For instance, to set batch size to 100,000:

```
C:\DBC Crane>dbcrane.cmd --config test_oracle_netezza.xml --batchsize 100000 TEST_RG1
```

### 8.4. Increasing number of threads

---

By default, a table is replicated in 2 threads. Increasing the number of threads would greatly improve performance for most database systems. But it would also increase the load on source database, target databases and the machine running DBCrane. System and database administrators will need to make sure these systems are not overloaded. When replication is run in multi threads, extraction would spread the load by rowid or partitions when applicable.



## 8.5. Configure JVM memory size

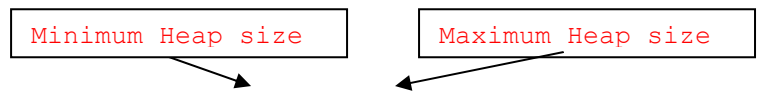
---

Increasing batch size or number of parallel threads would increase memory usage. The amount of memory available to DBCrane is limited by JVM (Java Virtual Machine) heap size. DBCrane would throw the following Java error when running out of JVM heap space.

```
java.lang.OutOfMemoryError: Java heap space
```

To increase JVM heap size, edit “dbcrane.cmd” in Windows or “dbcrane.sh” in Unix and save the changes:

- “dbcrane.cmd” in Windows



```
java.exe -Xms32M -Xmx512M -server -XX:+UseParallelOldGC -cp  
"%~dp0\DbCrane.jar;%~dp0\lib\*" us.bronzeage.dbcrane.Main %*
```

- “dbcrane.sh” in Unix

```
java -Xms32M -Xmx1024M -server -XX:+UseParallelOldGC -cp  
"$INSTALL_DIR/DbCrane.jar:$INSTALL_DIR/lib/*" us.bronzeage.dbcrane.Main "$@"
```

You can change size of JVM heap space by using Java command line options -Xms and -Xmx. Xms denotes minimum Heap size while Xmx denotes maximum Heap size.

For instance, to change DBCrane memory size to 128M (minimum) and 10G (maximum), modify the line beginning with “java.exe” in “dbcrane.cmd” to the following and save the changes:

```
java.exe -Xms128M -Xmx10240M -server -XX:+UseParallelOldGC -cp  
"%~dp0\DbCrane.jar;%~dp0\lib\*" us.bronzeage.dbcrane.Main %*
```

Make sure the machine has sufficient memory (RAM) before increasing JVM heap size. You might see the following error while starting DBCrane if the machine is running out of memory. To address this issue, you can reduce JVM heap size or add more memory to the machine.

```
Invalid maximum heap size: -Xmx512M  
The specified size exceeds the maximum representable size.  
Could not create the Java virtual machine.
```

## 8.6. Debugging mode

---

When running DBCrane in debugging mode, you can see memory usage on screen or in log file (when --log option is used).

```
C:\DBC Crane>dbcrane.cmd --config test_oracle_netezza.xml --debug true TEST_RG1  
...  
##### JVM Heap utilization statistics [MB] #####  
Used Memory: 667 (67%)  
Free Memory: 314
```

**DBC Crane User's Guide**  
**Version 2.0.0**

Total Memory: 982  
Max Memory: 982  
...

## 9. Appendix

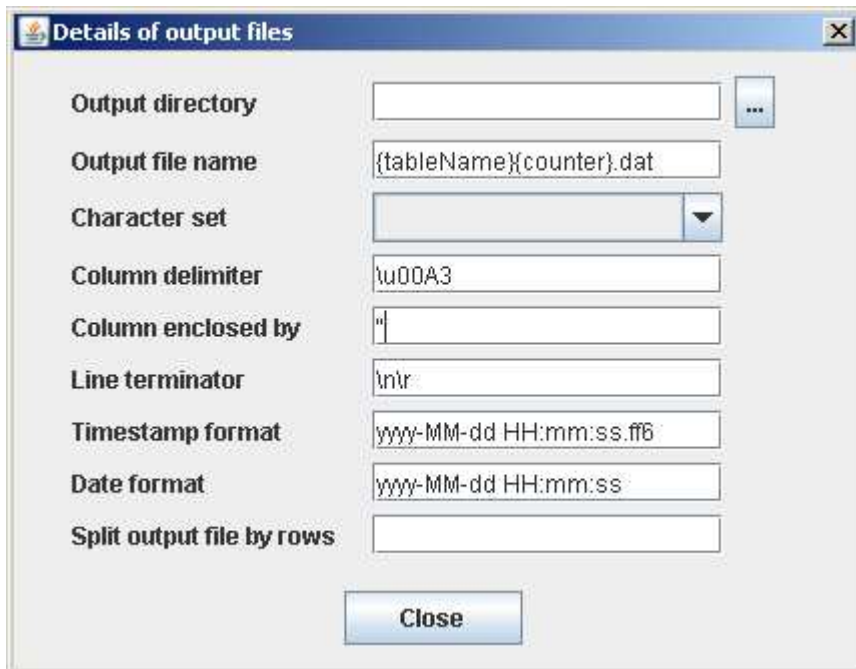
### Appendix A: Special characters

DBC Crane supports the following special characters:

\t	tab
\b	backspace
\n	newline
\r	carriage return
\f	formfeed
\uFFFF	FFFF is the Unicode value in hex, e.g. \u039A is the uppercase Greek omega character

### Examples

In the following example, line terminator is `\n\r`, column delimiter is the pound sign (`\u00A3`), column is enclosed by “ (double quotes).



## Appendix B: Date/Time format

The following pattern letters are defined

Letter	Date or Time Component	Presentation	Examples
G	Era designator	Text	AD
y	Year	Year	1996; 96
M	Month in year	Month	July; Jul; 07
w	Week in year	Number	27
W	Week in month	Number	2
D	Day in year	Number	189
d	Day in month	Number	10
F	Day of week in month	Number	2
E	Day in week	Text	Tuesday; Tue
a	Am/pm marker	Text	PM
H	Hour in day (0-23)	Number	0
k	Hour in day (1-24)	Number	24
K	Hour in am/pm (0-11)	Number	0
h	Hour in am/pm (1-12)	Number	12
m	Minute in hour	Number	30
s	Second in minute	Number	55
S	Millisecond	Number	978
z	Time zone	General time zone	Pacific Standard Time; PST; GMT-08:00
Z	Time zone	RFC 822 time zone	-0800

Pattern letters are usually repeated, as their number determines the exact presentation. FF6 is reserved for nanosecond.

### Examples

The following examples show how date and time patterns are interpreted in the U.S. locale. The given date and time are 2001-07-04 12:08:56 local time in the U.S. Pacific Time zone.

Date and Time Pattern	Result
"yyyy.MM.dd G 'at' HH:mm:ss z"	2001.07.04 AD at 12:08:56 PDT
"EEE, MMM d, ''yy"	Wed, Jul 4, '01
"h:mm a"	12:08 PM
"hh 'o''clock' a, zzzz"	12 o'clock PM, Pacific Daylight Time
"K:mm a, z"	0:08 PM, PDT
"yyyyy.MMMMM.dd GGG hh:mm aaa"	02001.July.04 AD 12:08 PM
"EEE, d MMM yyyy HH:mm:ss Z"	Wed, 4 Jul 2001 12:08:56 -0700
"yyMMddHHmmssZ"	010704120856-0700
"yyyy-MM-dd'T'HH:mm:ss.SSSZ"	2001-07-04T12:08:56.235-0700
"yyyy-MM-dd'T'HH:mm:ss.FF6Z"	2001-07-04T12:08:56.235774-0700

## Appendix C: Software License Key

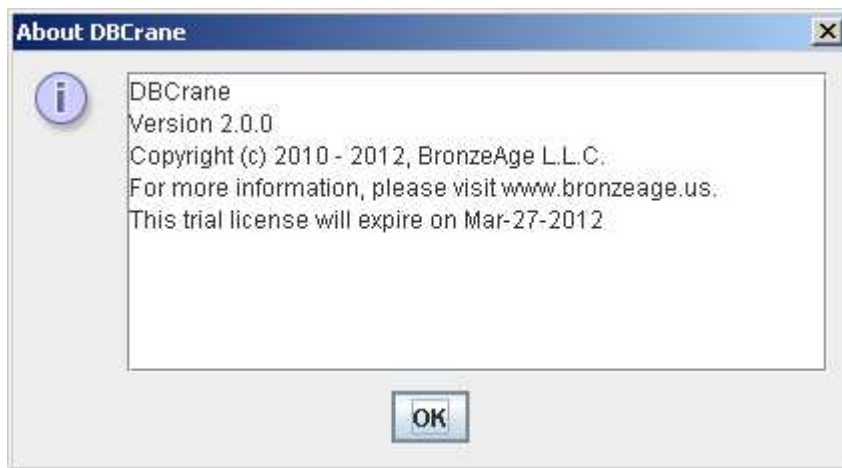
---

DBC Crane comes with a trial license key valid for 3 months. Please contact Bronzeage L.L.C. to obtain a permanent license key. License key file DbCrane.key is located in DBC Crane install directory.

## Appendix D: Viewing the version

---

Click Help -> About in the menu to open the About DBC Crane window. Software version and other information are displayed here.



You can also run DBC Crane in command line with `-h` option for version information.

```
C:\DBC Crane>dbcrane.cmd -h
```

```
DBC Crane  
Version 2.0.0  
Copyright (c) 2010 - 2012, BronzeAge L.L.C.  
For more information, please visit www.bronzeage.us.  
This trial license will expire on Mar-27-2012
```