Database Transfer

Version 4.5.1

Jack Gerrard

January 2018

Foreward

The focus of this document is the data transfer from a previous version database to the current uniCenta oPOS 4.5 database schema format.

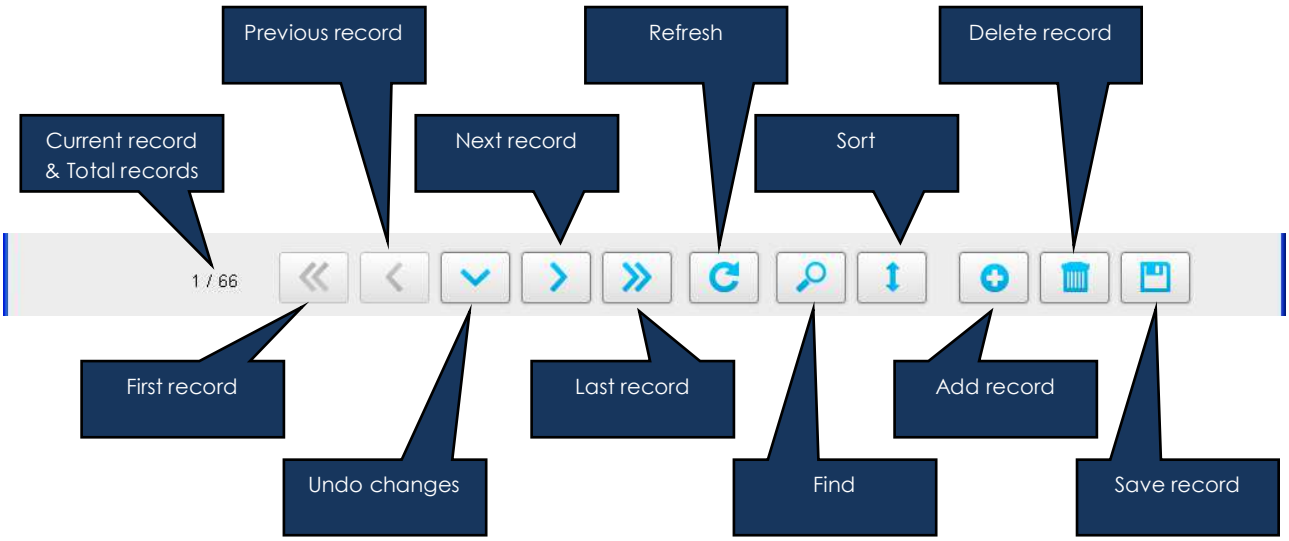
Any uniCenta oPOS Beta version should not be used in a live production environment

Date	Version	Author	Comment
June 2016	Beta1	Jack Gerrard	Draft
July 2016	Beta2	Jack Gerrard	Update
August 2016	Beta3	Jack Gerrard	Update & Final
September 2016	Beta4	Jack Gerrard	Update & Final
October 2016	4.2	Jack Gerrard	Final
February 2017	4.2	Jack Gerrard	Update & Final
May 2017	4.3.1	Jack Gerrard	Update & Final
September 2017	4.4.2	Jack Gerrard	Update
January 2018	4.5.1	Jack Gerrard	Update & Final

Useful things

Editor Toolbar

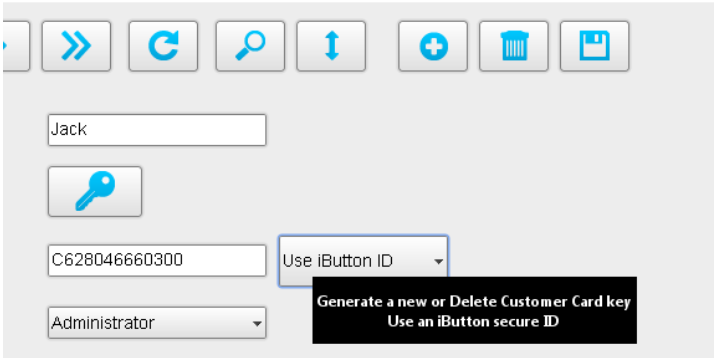
You will see this toolbar is used throughout uniCenta oPOS so here's a quick overview of what the buttons do. The Editor Toolbar appears in all record management forms.



Tooltips

Lots of things in uniCenta oPOS use pop-up Tooltips.

All you have to do is hover your mouse pointer over a component for a second and it will give you a brief description of what it does. Tooltips are constantly being added or revised as uniCenta oPOS development progresses.



TRANSFERS

In versions prior to v4 upgrading uniCenta oPOS it was a matter of stepping through each new release. So, for example if you were a user of v3.70 and wanted to move to 3.91.3 you had to download and install 3.8, then 3.9, 3.91.1, 3.91.2 and finally 3.91.3

uniCenta oPOS 4 changed all that by including new functionality to transfer data from Apache Derby Embedded, MySQL and PostgreSQL databases to the latest uniCenta oPOS version.

From version 4.5 we are gradually changing the way upgrades happen. In the past uniCenta oPOS releases used what is called the dot sequence and went something like this; *major-release.feature-set.bug-fixes* (4.4, 4.4.1, 4.4.2) where each iteration required a new full installer.

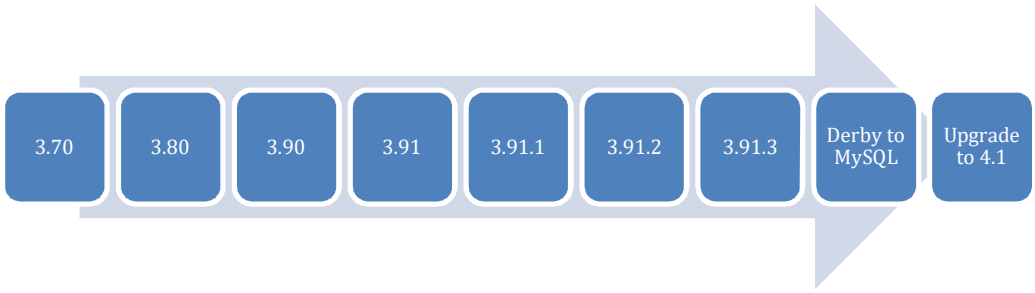
In the future users will only need to apply an update containing ***.bug-fixes*** which are applied to the ***major-release.feature-set*** and not require the full installer.

> **Example:**
> Currently using v4.4.2 – use the Database Transfer tool
> Currently using v4.5 – uses auto-Update

You do not have to use the Database Transfer tool to upgrade from v4.5 to v4.51(*unless you choose to*) Importantly

The Database Transfer tool does not make any changes whatsoever to your existing database; it just reads it.
It "imports" data from the source (*your previous version*) into its new database schema rather than making changes to the existing source schema. The original database source data is preserved, remains unaffected.

OLD WAY



NEW WAY

OVERVIEW

STEPS:

CREATE A NEW DATABASE SCHEMA

Create a new MySQL database (schema) – **unicentaopos** is used as an example throughout this guide - and any User Permissions if required.

*If you already have a database schema named **unicentaopos** choose another name for example: **unicentaopos45***

Refer to the MySQL manuals or use a tool like MySQL Workbench how to create a database schema. Make a note of the name. You need this in the next step along with the MySQL Username and Password. See our video page and watch the Workbench video

INSTALL 4.5 & CONFIGURE UNICENTA OPOS

After installing start uniCenta oPOS 4.5 and go to ***System>Configuration>Database Setup***.

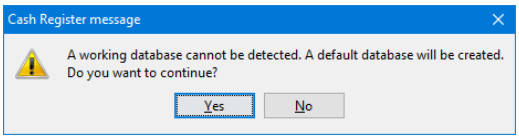
SWITCH TO NEW DATABASE SCHEMA

On the ***Database Setup*** tab: **Test** the connection to the MySQL server and select the database schema (**unicentaopos**) you created in Step-1 from the dropdown list.

SAVE, EXIT & RESTART

On restart you will receive a pop-up message to create a new database.

Answer **Yes**



Log in as Administrator – the ***Database Transfer*** option should only be run by the Administrator

Select and run the ***Utilities>Tools>Database Transfer***

PREPARATION

The key to a successful transfer is to run it on a sufficiently capable machine. Running the Database Transfer tool on a low-specification machine is not recommended.
Data transfer can be a computer intensive process and you need to make sure your machine has the capability to do this.

Please take a look at the **TEST SCHEDULE** section at the end of this document. It provides a useful guide to help you prepare for a successful database schema transfer.

MySQL's default packet setting is 4MB which is fine for a text only database transfer but it will quickly run out of steam where there are a significant number of images and or the accumulated size exceeds that limit.
It is essential you make a MySQL setting change to the **max_allowed_packet** – see below **how to change** – if you have used Product images.

Do not attempt to run the Database Transfer tool on a database if it is hosted remotely such as with a web hosting company. You should only run on a locally hosted database or on your own internal network machine.

If your database is remotely hosted; Make a copy (mysqldump) using a database admin tool such as MySQL Workbench (Export) and run the Database Transfer locally. Once complete, copy back to the remote host and Import into your hosted server.

If you are running uniCenta oPOS on several terminals which connects to a central database server run the Database Transfer tool from a capable machine first and then upgrade uniCenta oPOS on each of the other machines. You do not need to run the Database Transfer tool again.

IMPORTANT:
As always it is sensible to take a Backup of

uniCenta oPOS does not support MariaDB.

The following source database tables are not transferred during the Database Transfer process:

1. CSVImport – as this is transient data it is omitted
2. Resources – changes to scripts like i.e.: Printer.Tickets will need to be redone
3. Roles – new Permissions have been added
4. Any user modified, non-standard table or table column – these will likely cause error messages and prevent the transfer FROM completing

If you receive an error message - similar to the snippet below - or the transfer process appears to 'freeze' for a long period the transfer has very likely to have failed and you will need to restart from Step #1 after following the **How to fix** section below.

```
com.mysql.jdbc.PacketTooBigException:

Packet for query is too large (5098311 > 4194304). You can change this value
 on the server by setting the max_allowed_packet variable.
```

How to fix:
Follow the uniCenta oPOS Database Transfer Guide and create a new schema using a DB Admin tool – like Workbench – and immediately run this command against the new schema;

```
SET GLOBAL max_allowed_packet=50*1024*1024
```

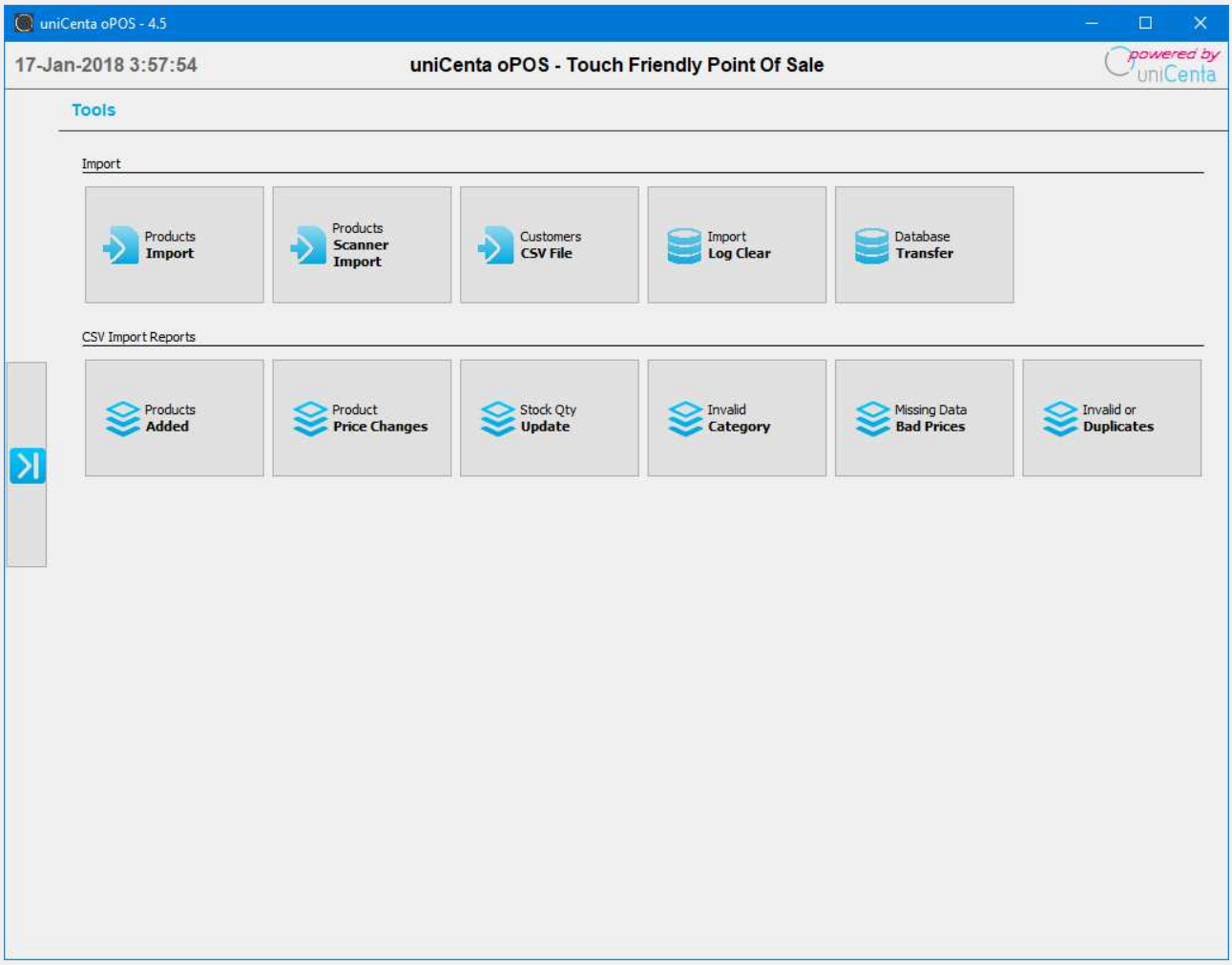
The 50 value is just an example and you may have to experiment to get the right value as it depends on the content of the database to transfer FROM.
Start uniCenta oPOS and run the Database Transfer tool against the new schema.

You can, of course, set this as a MySQL default in your my.cnf (Linux and Mac) or my.ini (Windows). Please refer to the MySQL documentation how to do this.

If your MySQL database is located with a web hosting company you may need to discuss with them how to set the MySQL configuration.

An online "How To" video is available https://unicenta.com/pages/upgrade-unicenta-opos/

DATABASE TRANSFER



The Database Transfer option is accessed from the pop-out main menu under ***Utilities>Tools>Database Transfer***

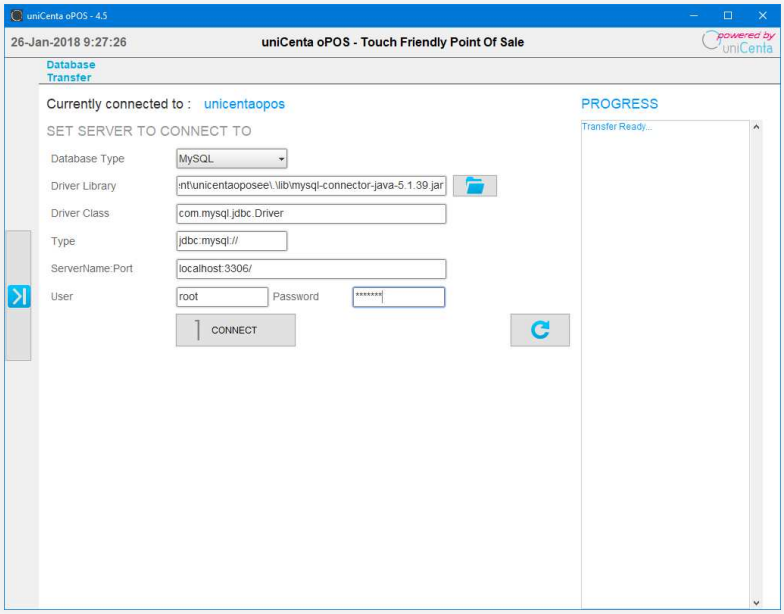
TRANSFER

The transfer process is a simple 1-2-3 step process

1 – Connect to the database source server

2 – Select the database to transfer FROM

3 – Start the transfer



1 – Connect to the database source server

Currently connected to :

Uses the current session's database '**unicentaopos**' As specified in PREPARATION : Step 1

PROGRESS

Activity list of each process

Database Type

Choose the database type to transfer FROM. Options available are Apache Derby, MySQL, PostgreSQL

Default : MySQL

Driver Library

Automatically filled when selecting Database Type. No need to change unless absolutely necessary.

Default : C:\Program Files (x86)\unicentaopos-4.5\.\lib\mysql-connector-java-5.1.39.jar

Driver Class

Automatically filled when selecting DatabaseType. No need to change unless absolutely necessary.

Default : com.mysql.jdbc.Driver

Type

The database connection type

Default : jdbc:mysql:// (*must end with* //)

ServerName:Port

The Server or IP address

Default : localhost:3306/ (*must end with* /)

Options:If using Linux or Mac you will likely need to change **localhost** to 127.0.0.1

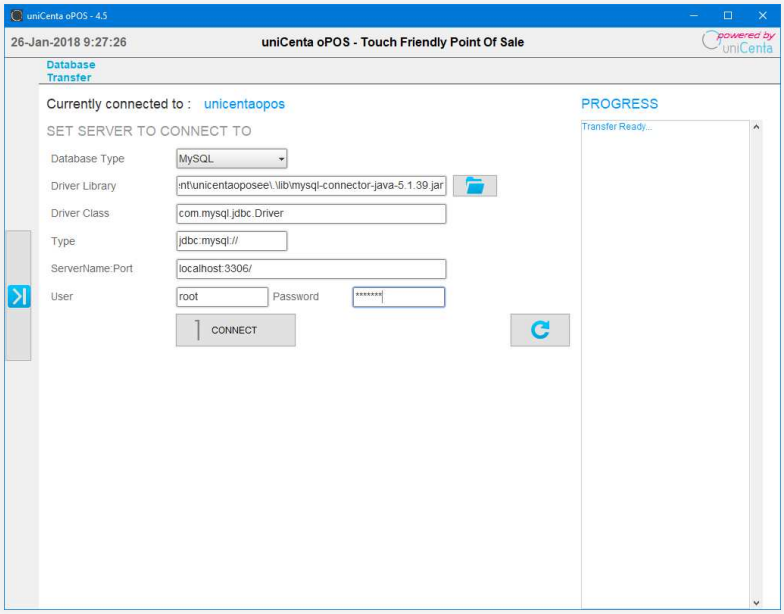
User

The User name to access the source database. It is recommended you use a MySQL User which has full permissions for example: **root**
Generally Apache Derby Embedded databases are set without the need for a Username. MySQL and PostgreSQL require a User name.

Password

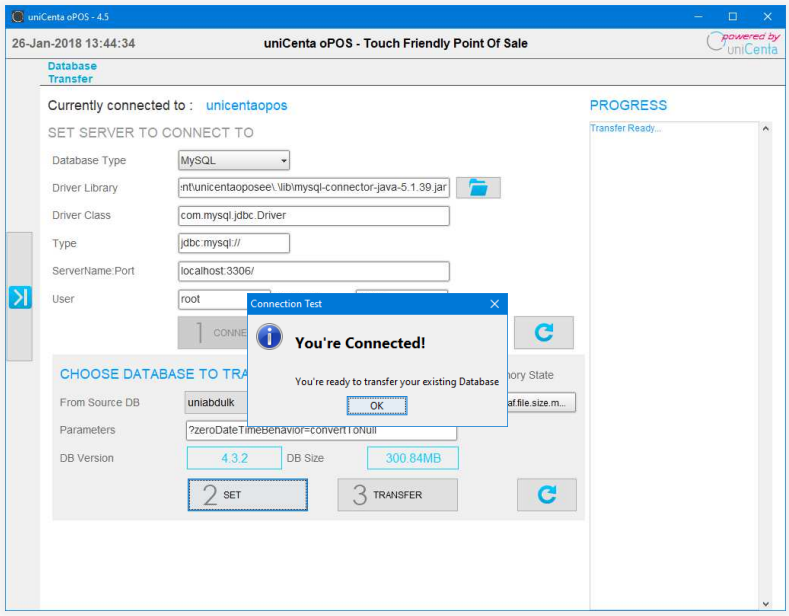
The password for the **User** specified above.
Generally Apache Derby Embedded databases are set without the need for a password. MySQL and PostgreSQL require a Password.

With all the above completed press the 1 CONNECT button to connect the Server



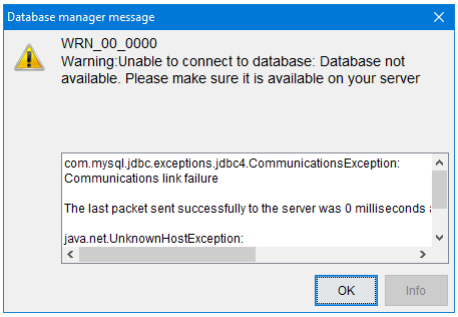
You can reset the form to its default settings by clicking the C button.

On successful connection to the Server; a pop up message will appear. **You're Connected!**

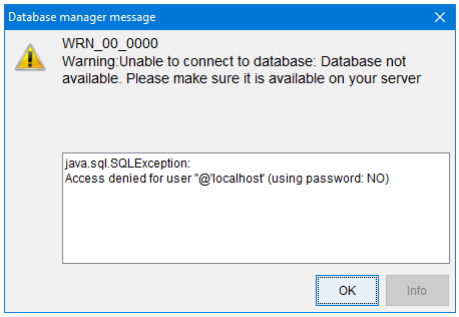


A lower panel will appear and you can now **CHOOSE THE DATABASE TO TRANSFER FROM**

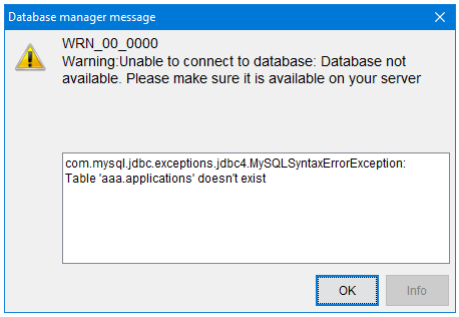
POSSIBLE ERROR MESSAGES



The ServerName:Port is incorrect

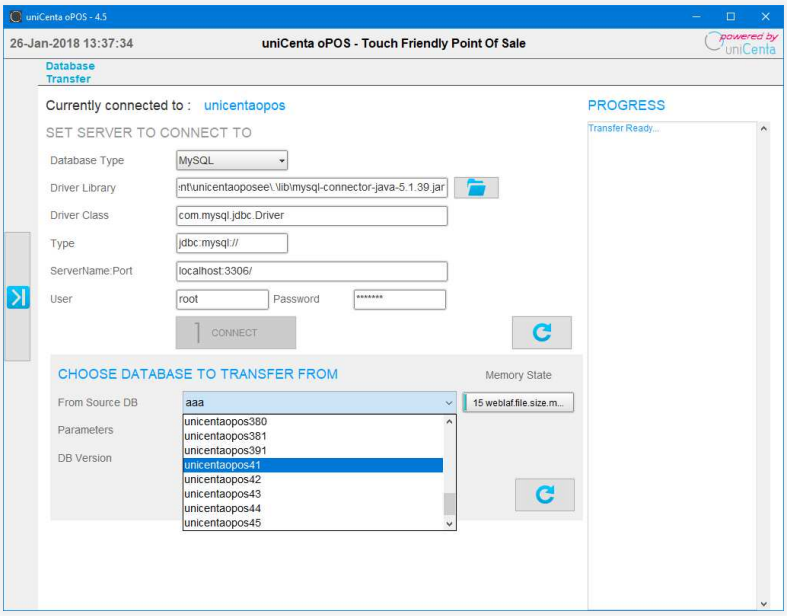


Incorrect User and/or password



Database doesn't exist

2 SET the database to transfer FROM



From Source DB

The dropdown list displays all the database schema available on the connected Server. Choose the database you created earlier.

Parameters

You can specify additional MySQL connection options as required.

Default: ?zeroDateTimeBehavior=convertToNull

Having chosen the database schema to transfer FROM press the 2 SET button.

Information from the source database is displayed:

DB Version - The version of the **From Source DB**

DB Size - The size in MB of the **From Source DB**

At this point is strongly recommended you compare the **DB Size** value and the capability of your machine – see **TEST SCHEDULE** and the **COMMAND LINE** pages at the end of this document.

You can safely exit the Database Transfer tool at this point if required.

To clear any changes to the form's lower panel; click the ↻ button to restore its default settings.

3 Start the TRANSFER

During the transfer process you will receive dynamic feedback from:

The PROGRESS List

Dynamically updated list of process events as they happen

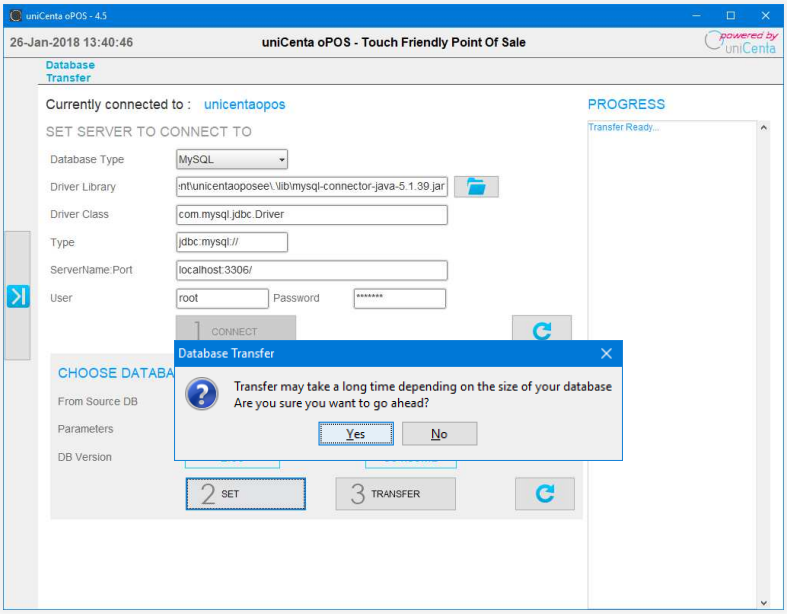
The Memory State Bar

An animated RAM state bar updated every second.
Hover mouse over the bar to show Allocated, Max and Current RAM usage

The Progress Bar

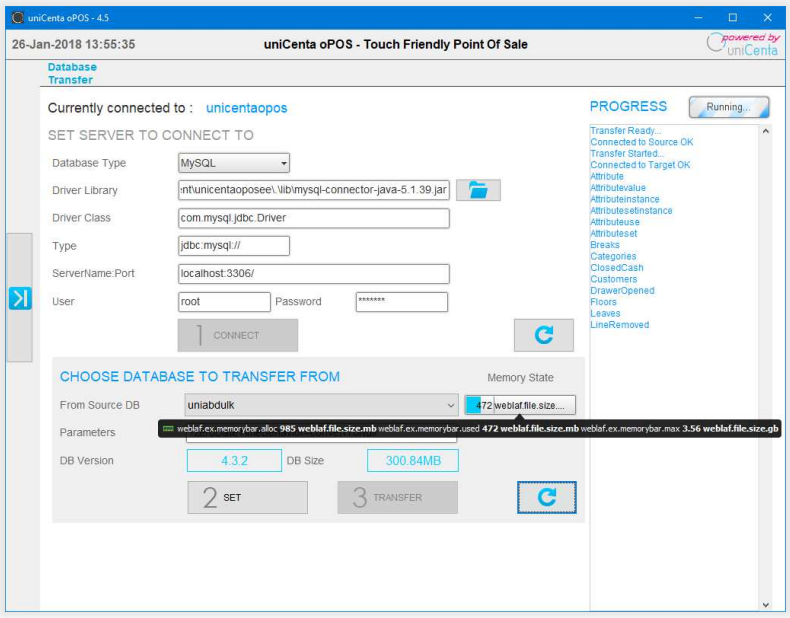
Shows four main processing steps:

1. Starting - Getting things ready…
2. Running - Transferring your data…
3. Foreign Keys - As it says...adding foreign keys & etc…
4. Finished - Steps 1 – 3 completed successfully



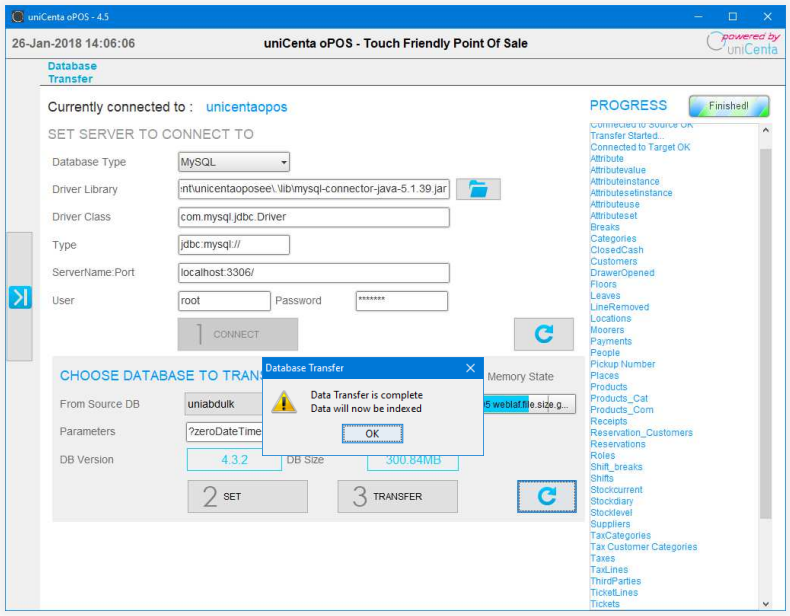
You have the choice to continue with the database transfer or not.

DATA TRANSFER IN PROCESS

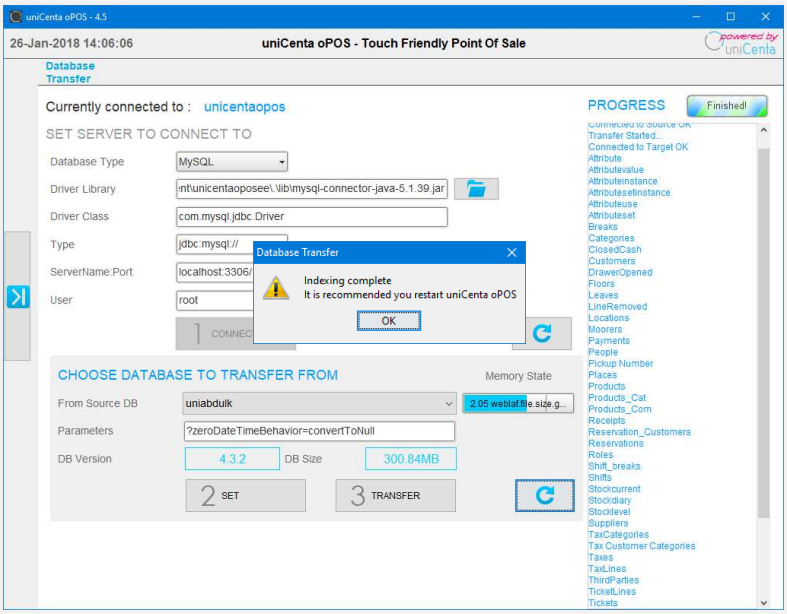


The Progress Bar may pause from time-to-time for several reasons for example; whilst the machine's hard disk writes the transfer data.
If it stops for an undue length of time; check the Memory Bar. If it is mostly blue then your machine has possibly reached the JVM's Maximum RAM settings and you will have to restart the Database Transfer from Step-1

DATA TRANSFER COMPLETED



INDEX DATA PROCESS



When the transfer is complete the newly transferred data is immediately available and ready for use, but it is recommended you restart uniCenta oPOS

Note: If the source database is very large – see the Memory State – it may have consumed a high percentage of available (JVM) memory during the transfer process.
If it is beyond 50% it is recommended you restart uniCenta oPOS else performance may be affected.

After restarting uniCenta oPOS it is advisable to check that everything is OK and all the data transferred successfully.

TEST SCHEDULE

RIG

Machine	HP Compaq 8100 Elite SFF
OS	Windows 10 Pro
Type	x64
Processor	Intel Core i7 @2.80GHz
RAM	12GB
Disk	450GB SSD

uniCenta oPOS DATABASE TRANSFER TEST SCHEDULE

Source DB	Source Vers.	DB	DB Size	RAM Used	Start Hr	End Hr	Elapsed	Suggested min. JVM
*Openbravo	2.3.0	MySQL	984.71MB	7.1GB	12:53:00	14:06:00	01:13:00	-Xms256m -Xmx10248m
uniCenta	4.3.2	MySQL	302.21MB	4.12GB	14:30:00	14:39:02	00:09:02	-Xms128m -Xmx5124m
uniCenta	4.5.0	MySQL	241.30MB	3.85GB	07:21:10	07:26:22	00:05:12	-Xms256m -Xmx4096m
uniCenta	4.2.2	MySQL	167.57MB	3.39GB	15:16:30	15:19:55	00:03:25	-Xms256m -Xmx4096m
uniCenta	3.91.3	Derby	92.00MB	1.18GB	15:47:45	15:48:17	00:00:32	-Xms128m -Xmx2048m
uniCenta	4.3.2	MySQL	74.10MB	506MB	15:27:00	15:27:15	00:00:15	-Xms128m -Xmx1024m
uniCenta	4.1.1	MySQL	24.64MB	458MB	15:06:00	15:06:12	00:00:12	-Xms64m -Xmx1024m
uniCenta	4.4.2	MySQL	3.87MB	144MB	15:35:35	15:35:40	00:00:05	-Xms64m -Xmx1024m

NOTES:

All databases tested are from real uniCenta customers and live sites
Processing times will vary depending on machine **TYPE** and **PROCESSOR** specifications
SSD **Disks** are generally a lot quicker than mechanical disks
MySQL PROCESSING latency - WRITING to disk will continue for some minutes after the Database Transfer completes. Check your machine's HDD activity indicator. Write speed is dependent on the combination of **MySQL** settings & **Disk capability.**

EXTRACT:

*** Source DB** Openbravo – 10 tables from a total of 37

TABLES	RECORDS
categories	27
closedcash	9255
payments	221564
products	912
products_cat	453
receipts	221350
stockdiary	453037
taxlines	250845
tickets	221319
ticketlines	516801

1,644,718 records out of 1,864,226 total

COMMAND LINE

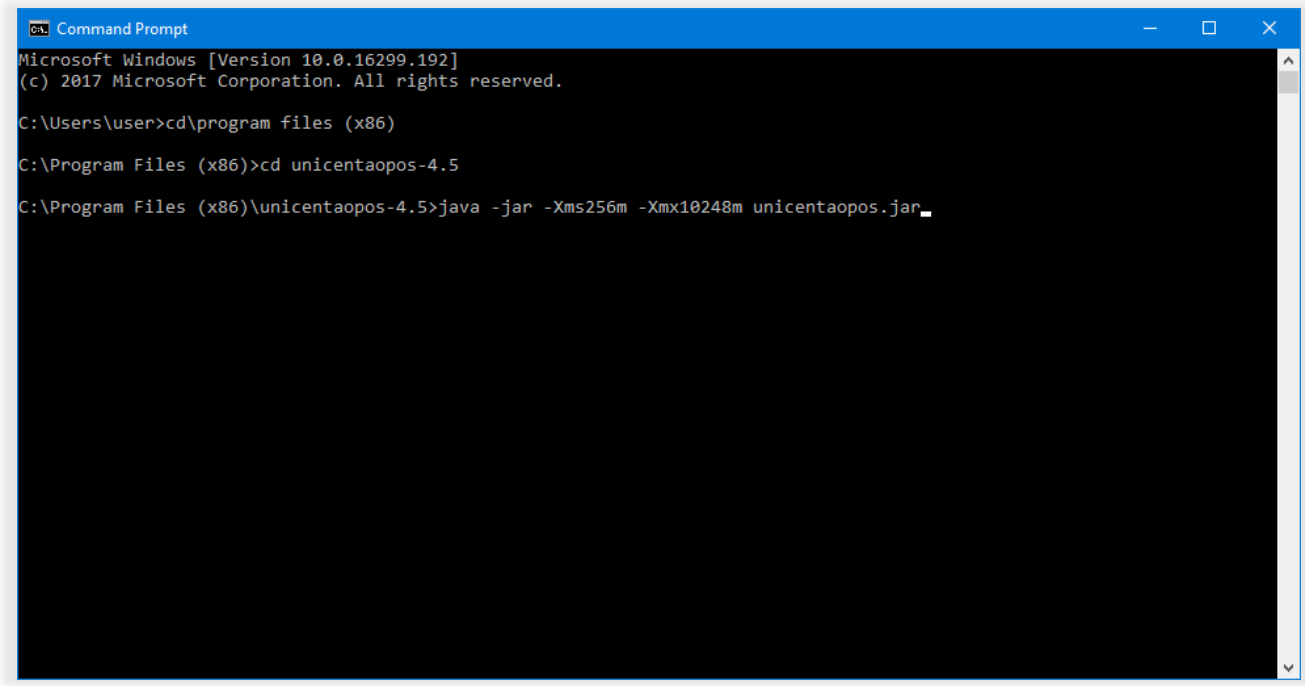
Windows

uniCenta oPOS is launched using a batch file named **start.bat** - it contains commands which are fine for standard use, but you may need to change before running the Database Transfer tool depending on the size of your database.

You can run uniCenta oPOS from the Command line and adjust memory allocation settings by changing the JVM **-Xms** and -**Xmx** values.

The standard uniCenta Opos **start.bat**

```
start /B javaw -Xms512m -Xmx1024m -cp %CP% -
Djava.library.path="%DIRNAME%lib/Windows/i368-mingw32" -Ddirname.path="%DIRNAME%./" -
splash:unicenta_splash_dark.png com.openbravo.pos.forms.StartPOS %1
```



A Windows Command line example where the database to transfer FROM size is just under 1GB on a machine which has 12GB of RAM:

```
java -jar -Xms256m -Xmx10248m unicentaopos.jar
```

Your settings should always allow 1MB spare for your Operating System if you change the **-Xm** settings.

Linux / Mac OS X

The standard uniCenta Opos **start.sh**

```
java -cp $CP -Xms512m -Xmx1024m -splash:unicenta_splash_dark.png -
Djava.library.path=$DIRNAME$LIBRARYPATH -Ddirname.path=$DIRNAME/
com.openbravo.pos.forms.StartPOS
```