

openCRX Installation Guide for MaxDB 7

Version 1.4.0

www.opencrx.org

openCRX Installation Guide for MaxDB 7: Version 1.4.0

by www.opencrx.org

The contents of this file are subject to a BSD license (the "License"); you may not use this file except in compliance with the License.

You may obtain a copy of the License *here* (<http://www.opencrx.org/license.htm>)

Table of Contents

1. About this Book	1
Who this book is for	1
What do you need to understand this book	1
2. Prerequisites	2
3. Upgrading from previous versions	3
Upgrading from ASCII to UNICODE	3
4. Create the database	6
5. Configure SQL User	14
6. Configure Backup	15
7. Install Database Schema	18
8. Next Steps	21
A. Appendix	22
Bibliography	23

List of Figures

2-1. Select the installation profile	2
3-1. Create the Database Instance	3
4-1. Create the Database Instance	6
4-2. Select a Database Template	6
4-3. Set the database name	7
4-4. Select installation version	7
4-5. Select the startup mode	8
4-6. Set the database manager login	8
4-7. Set the database configuration values	9
4-8. Set the database instance type	9
4-9. Set the database parameter values	10
4-10. Specify the data and log values	11
4-11. Specify the backup volume	11
4-12. Set the database administrator login	12
4-13. Create the database instance	12
4-14. Progress of database instance creation	13
4-15. Creation of database instance successful	13
5-1. Start the SQL User management wizard	14
5-2. Create an SQL User	14
6-1. Start the backup configuration wizard	15
6-2. Select or create a data backup media	15
6-3. Start data backup	16
6-4. Start the backup wizard	16
6-5. Select or create a log backup media	17
6-6. Activate automatic log backup	17
7-1. Start SQL Studio	18
7-2. Login to SQL Studio	20
7-3. Open an SQL Dialog and run openCRX schema creation scripts	20
7-4. Verify the created database objects	20

List of Examples

3-1. Copying data to a new database instance	4
--	---

Chapter 1. About this Book

This book describes how to setup an *openCRX* database instance for *MaxDB*.

Who this book is for

The intended audience are *openCRX* database administrators.

What do you need to understand this book

This book describes the installation of *openCRX* for *MaxDB*. The book assumes that you are familiar with *MaxDB* installation and configuration.

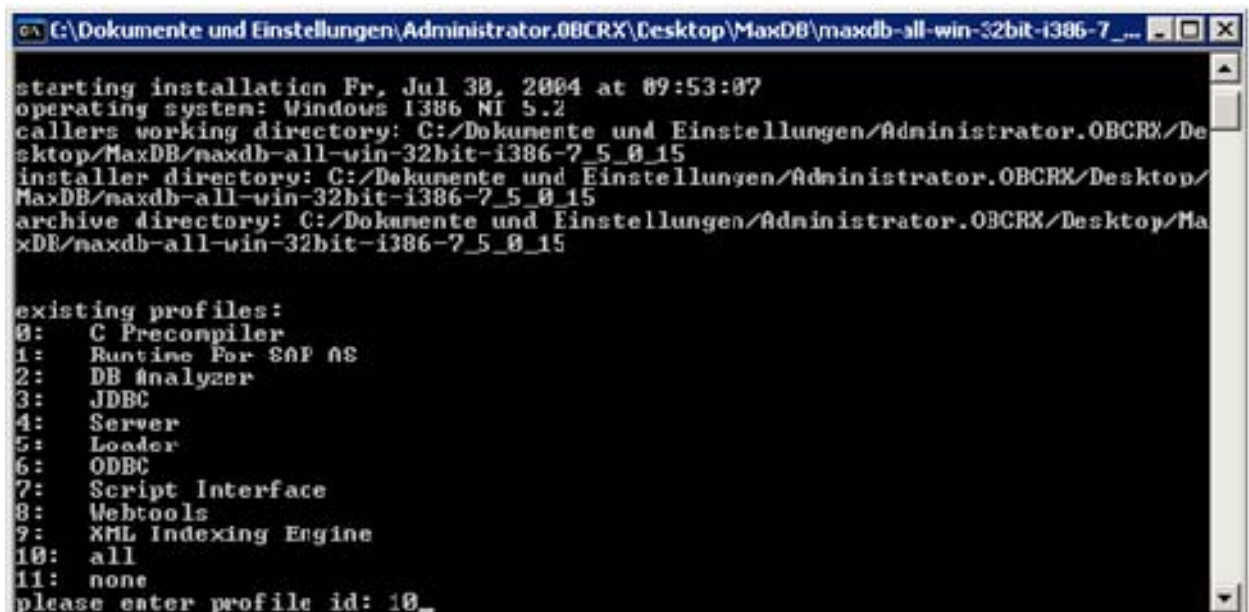
Chapter 2. Prerequisites

As a first step you must download the following software packages:

- Download **openCRX for MaxDB** from *here* (http://sourceforge.net/project/showfiles.php?group_id=95219) (e.g. *opencrx-1.4.0-core.maxdb-7.zip*). The distribution contains the *MaxDB* scripts required to install the *openCRX* database.
- Download **MaxDB Server** from *here* (<http://dev.mysql.com/downloads/maxdb/7.5.00.html>).
- Download **MaxDB SQL Studio, MaxDB DBM GUI, MaxDB JDBC Driver** from *here* (<http://dev.mysql.com/downloads/maxdb/clients.html>).

As a next step you must install *MaxDB*. The chapter **Installation -> Database Software Installation Guide -> Standard Installation with SDBINST** from *MySQL MaxDB Library* (<http://dev.mysql.com/doc/maxdb/en/default.htm>) explains in detail how to install the database. Choose the installation profile **all** as shown in *Figure 2-1*.

Figure 2-1. Select the installation profile.



```
C:\Dokumente und Einstellungen\Administrator.OBCRX\Desktop\MaxDB\maxdb-all-win-32bit-i386-7_...
starting installation Fr, Jul 30, 2004 at 09:53:07
operating system: Windows I386 NT 5.2
callers working directory: C:\Dokumente und Einstellungen\Administrator.OBCRX/De
sktop/MaxDB/maxdb-all-win-32bit-i386-7_5_0_15
installer directory: C:\Dokumente und Einstellungen\Administrator.OBCRX/Desktop/
MaxDB/maxdb-all-win-32bit-i386-7_5_0_15
archive directory: C:\Dokumente und Einstellungen\Administrator.OBCRX/Desktop/Ma
xDB/maxdb-all-win-32bit-i386-7_5_0_15

existing profiles:
0: C Precompiler
1: Runtime For SAP AS
2: DB Analyzer
3: JDBC
4: Server
5: Loader
6: ODBC
7: Script Interface
8: Webtools
9: XML Indexing Engine
10: all
11: none
please enter profile id: 10_
```

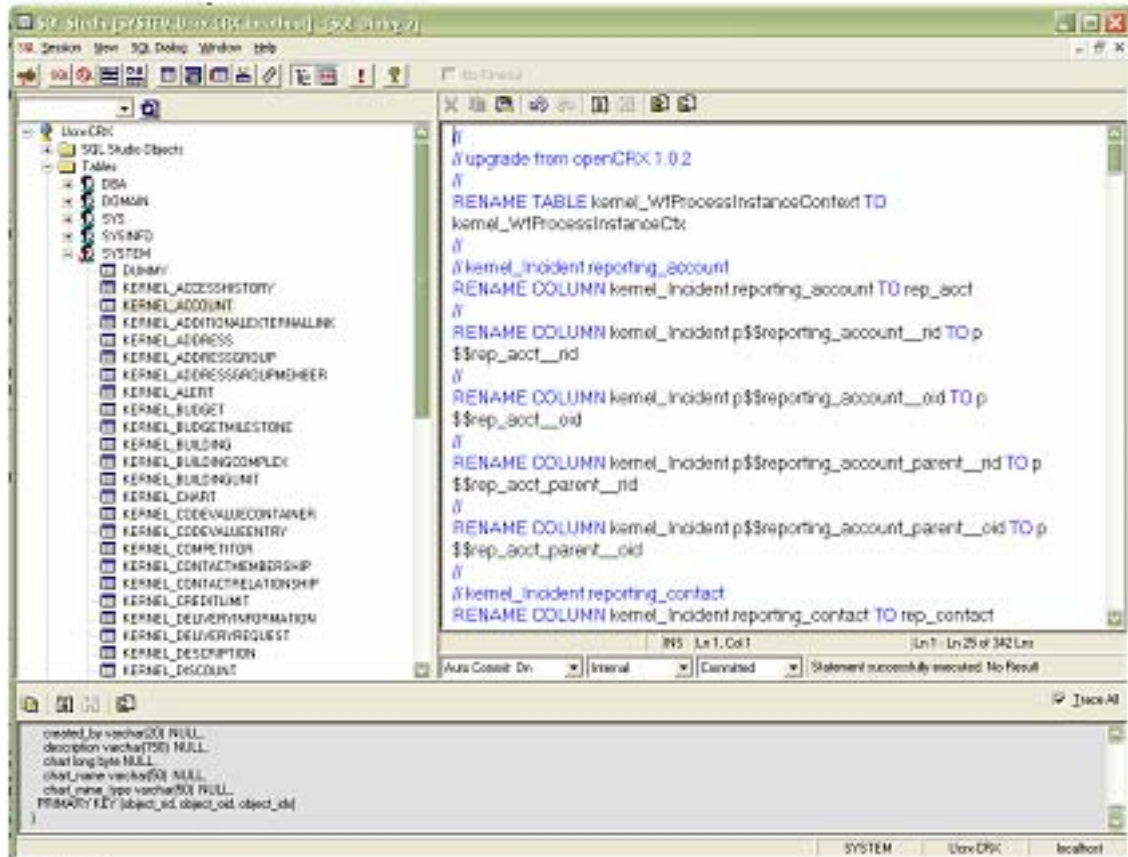
This document assumes that you use the **Database Manager GUI (DBM GUI)** and **SQL Studio for Windows** for database administration. *DBM GUI* and *SQL Studio* come with a Windows installer. The installation is straightforward. The JDBC Driver is required for the application server installation.

Chapter 3. Upgrading from previous versions

If you already have *MaxDB* for *openCRX* installed, upgrade the database as explained below. You can then skip the rest of this document.

The *openCRX* distributions provide an SQL script of the form *upgrade-from-<version from>-to-<version to>.sql*. E.g. If you have installed *openCRX* 1.3.0 and you want to upgrade to version 1.4.0 you have to run the script *upgrade-from-1.3.0-to-1.4.0.sql* on your database instance.

Figure 3-1. Create the Database Instance.



You can run the script as follows:

- Open *SQL Studio* and login to the database instance you want to upgrade.
- Copy/paste the upgrade script into an SQL window of *SQL Studio* and execute the script as shown in *Figure 3-1*.
- If you get error messages it is likely that you have either run the script on a database instance of the wrong version or you have run the script more than once.

Upgrading from ASCII to UNICODE

This section explains how to upgrade an ASCII database instance (`_UNICODE=NO`) to a UNICODE (`_UNICODE=YES`) database instance. Unfortunately there is no easy way to do this migration. We describe two solutions where both of them have their disadvantages as explained below.



Warning Please note that the row size of UNICODE tables is limited to 4,000 characters with MaxDB (as opposed to 8,000 chars with ASCII tables). A row size of 4,000 characters is very limiting and you will run into various problems running openCRX on such a DB instance. We do not recommend running openCRX with MaxDB in UNICODE mode for this reason. All MaxDB scripts provided with openCRX are designed to create ASCII tables. You need to change these scripts to create UNICODE tables.

Solution A: loadercli

The *MaxDB* command-line utility *loadercli* allows to save an existing database instance and restore it into a new instance. In the following sample it is assumed that you have an existing database *crx-CRX* with *ASCII* encoding and a new and empty database *Ucrx-CRX* with *UNICODE* encoding. First, the data is exported from *crx-CRX* and then loaded into *Ucrx-CRX*. Using *loadercli* this can be done as follows:

Example 3-1. Copying data to a new database instance

```
>loadercli -d crx-CRX -u system,manager
Loader Server Protocol File: 'd:\pgm\sapdb\indep_data\wrk\loader.prt'

User SYSTEM connected to database CRX-CRX on local host.
loadercli>tableextract user data outstream file 'crx-CRX.data' records ucs2
OK

---
loadercli>quit
>loadercli -d Ucrx-CRX -u system,manager
Loader Server Protocol File: 'd:\pgm\sapdb\indep_data\wrk\loader.prt'

User SYSTEM connected to database UCRX-CRX on local host.
loadercli>tableload user data instream file 'crx-CRX.data' records ucs2
ERR -25500
Errors processing command exceed maximum number of errors to ignore; processing stopped

---
```



Important For some unknown reason we did not manage it to get the export/import process working without errors. Feedback is very welcome if somebody knows a solution.

Solution B: ODBC Client

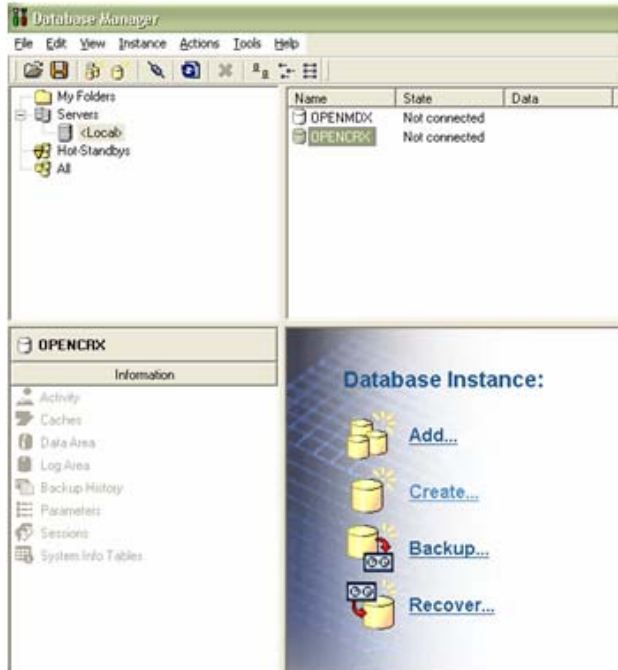
The databases can easily be migrated from ASCII to UNICODE (*crx-CRX* to *Ucrx-CRX*) with the use of an *ODBC* client which is able to handle two database sessions at the same time. E.g. you can use the *Microsoft SQL Server Enterprise Manager* and copy the database using the *Data Transformation Services -> Import Data / Export Data Wizard*. We do not know of an open source client doing the same job. Feedback is very welcome. Using the *DTS* wizard you can migrate the data as follows:

- Create two *ODBC* datasources. If you are working on an Windows operating system you can do this with *Administrative Tools -> Data Sources (ODBC)*. Add the first datasource *crx-CRX* with *MaxDB* as driver. Add the second datasource *Ucrx-CRX* with *MaxDB (Unicode)* as driver.
- Now start the *Microsoft SQL Server Enterprise Manager*. Create a new temporary database *crx-TMP*. Create the tables with the database create scripts for *SQL Server 2000*, e.g. *.../sql-server-2000/dbcreate-tables.sql*
- Start the *DTS Import/Export Wizard*.
- Select as source *MaxDB / crx-CRX*.
- Select as destination the database *crx-TMP*.
- Select all tables and deselect the views.
- Run the import. You should now have all your data in the database *crx-TMP*.
- Start the *DTS Import/Export Wizard*.
- Select as source the database *crx-TMP*.
- Select as destination the database *MaxDB (Unicode) / Ucrx-CRX*.
- Select all tables and deselect the views. By default the names of the destination tables are in double quotes and in upper/lower case and have the user name as prefix. Unfortunately, the export does not work with these default names. All names must be without quotes and all in uppercase. To change all the table names is a cumbersome work but you have to do it only once ;-)
- Run the import. You should now have all your data in the database *Ucrx-CRX*.

Chapter 4. Create the database

The *DBM GUI* allows you to create and manage databases. Before you can install the *openCRX* database schemas you must install a new database for *openCRX*. To do this start the *DBM GUI*. The startup screen looks as shown in *Figure 4-1*. Click **Create...** in the **Database Instance:** pane in order to create a new database instance.

Figure 4-1. Create the Database Instance.



You should now see the first page of the instance creation wizard as shown in *Figure 4-2*. Choose the template **Blank** and click **Next**.

Figure 4-2. Select a Database Template.



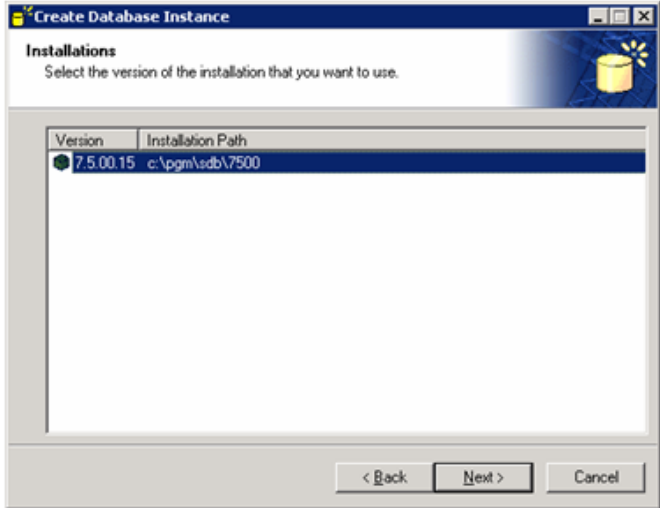
In the next step you must enter a name for the database instance as shown in *Figure 4-3*. Set the name to **crx-CRX**. Leave the field **Port** blank unless you are an experienced *MaxDB* user. The naming scheme **crx-*<instance>*** prevents you from naming conflicts in case you want to have multiple databases. The prefix **crx** is a unique prefix for all *openCRX* database instances. The suffix ***<instance>*** assigns a unique name of *openCRX* databases.

Figure 4-3. Set the datase name.



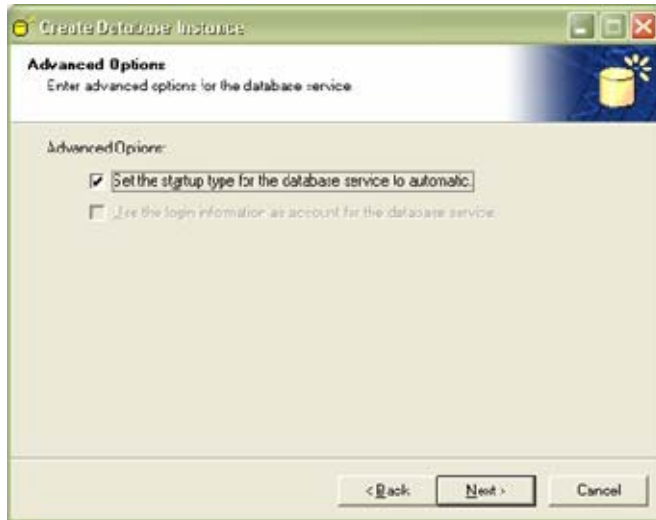
The next dialog *Figure 4-4* shows you the available *MaxDB* (and *SapDB*) versions and installation locations. *openCRX* has been tested with version **7.5.00.15**.

Figure 4-4. Select installation version.



The next dialog *Figure 4-5* allows you to setup the database instance as an automatic Windows service. If you activate this option the database is started automatically on Windows startup. If you do not activate this option you must start and stop the database instance using the *DBM GUI* or with shell scripts using the command line tools of *MaxDB*.

Figure 4-5. Select the startup mode.



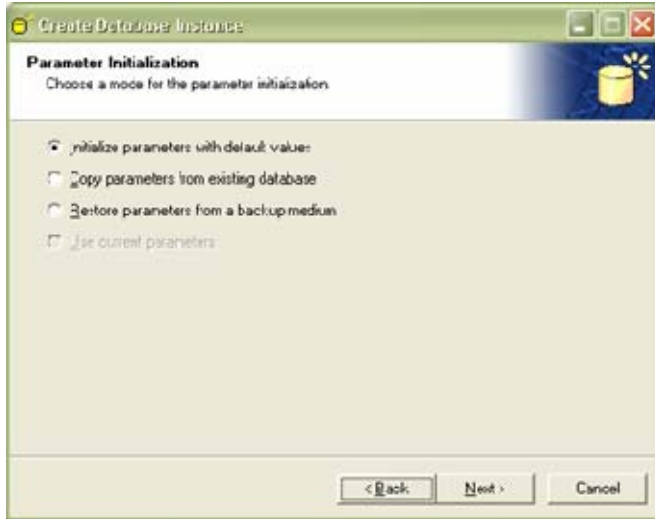
In the next dialog you specify the user name and password for the database manager as shown in *Figure 4-6*. It is strongly recommended that you choose the default values **dbm/dbm** at this time. You can change the values later on when the instance is up and running.

Figure 4-6. Set the database manager login.



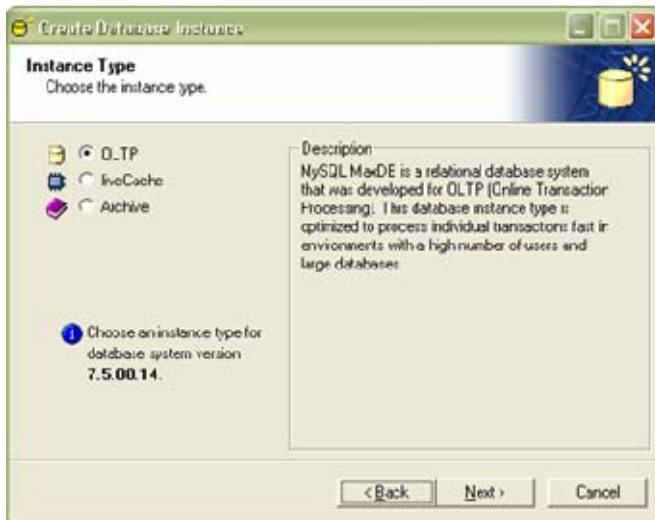
The next dialog *Figure 4-7* allows you to choose between using default or user-defined database configuration values. Unless you are an experienced *MaxDB* user we recommend to select the option **Initialize parameters with default values**.

Figure 4-7. Set the database configuration values.



In the next dialog *Figure 4-8* select the option **OLTP**. The database will be accessed by the application server in OLTP mode.

Figure 4-8. Set the database instance type.



The dialog *Figure 4-9* allows you to set various database options. The default options work fine with *openCRX*. Nevertheless, the option `_UNICODE` requires special mention.



Warning Please note that the row size of UNICODE tables is limited to 4,000 characters with MaxDB (as opposed to 8,000 chars with ASCII tables). A row size of 4,000 characters is very limiting and you will run into various problems running openCRX on such a DB instance. **We do not recommend running openCRX with MaxDB in UNICODE mode for this reason.** All MaxDB scripts provided with openCRX are designed to create ASCII tables. You need to change these scripts to create UNICODE tables.

The downside of running MaxDB in ASCII mode is that UTF-8 encoded multi-byte characters cannot be stored...

If you insist on running MaxDB in UNICODE mode even after reading the above warning, here is how to get it done. Select the tab **Extended** and then set the option `_UNICODE` to **YES**.



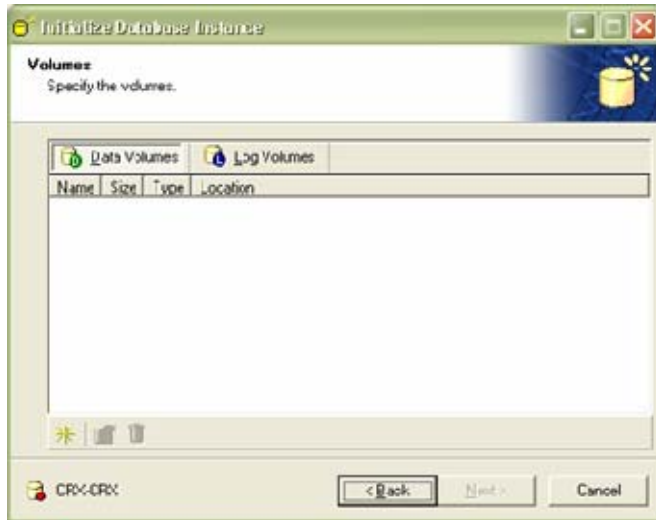
Important Once you have created the database instance, **MaxDB does NOT allow to change this option at a later time.**

Figure 4-9. Set the database parameter values.



In the next dialog *Figure 4-10* you must create volumes for **Data** and **Log**. Click the * Icon in order to create a new **Data** volume. Set the size to at least **200 MB** and click OK. Select **Log Volumes** and click the * Icon in order to create a new **Log** volume. Set the size to at least **200 MB** and click OK. The database size can be increased easily later on with the *DBM GUI* by adding additional **Data** and **Log** volumes.

Figure 4-10. Specify the data and log values.



You must now specify the **Backup** media for the created **Data** and **Log** volumes as shown in *Figure 4-11*. The default values are fine at this time. You can configure **Backup** media at a later time with the *DBM GUI*.

Figure 4-11. Specify the backup volume.



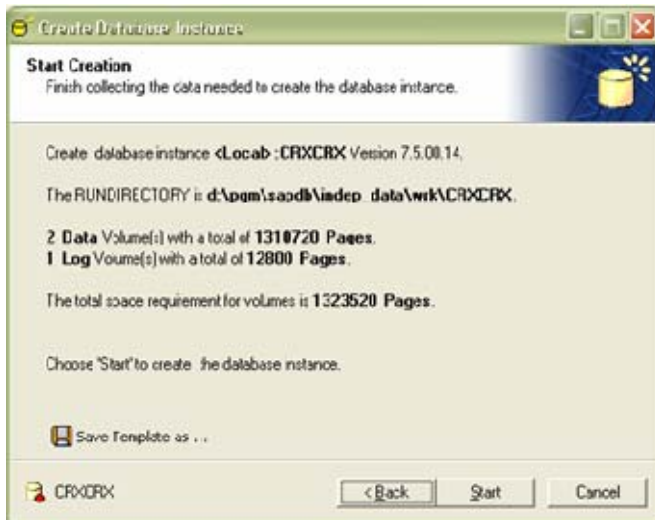
In the next step you must specify the name and password for the database administrator as shown in *Figure 4-12*. It is strongly recommended to set the default values to **dba/dba** at this time. You can change names and password at a later time.

Figure 4-12. Set the database administrator login.



Finally the summary dialog shows the configured values before the database creation process is started as shown in *Figure 4-13*.

Figure 4-13. Create the database instance.



The database creation can take a while. You should see a progress dialog as shown in *Figure 4-14*.

Figure 4-14. Progress of database instance creation.



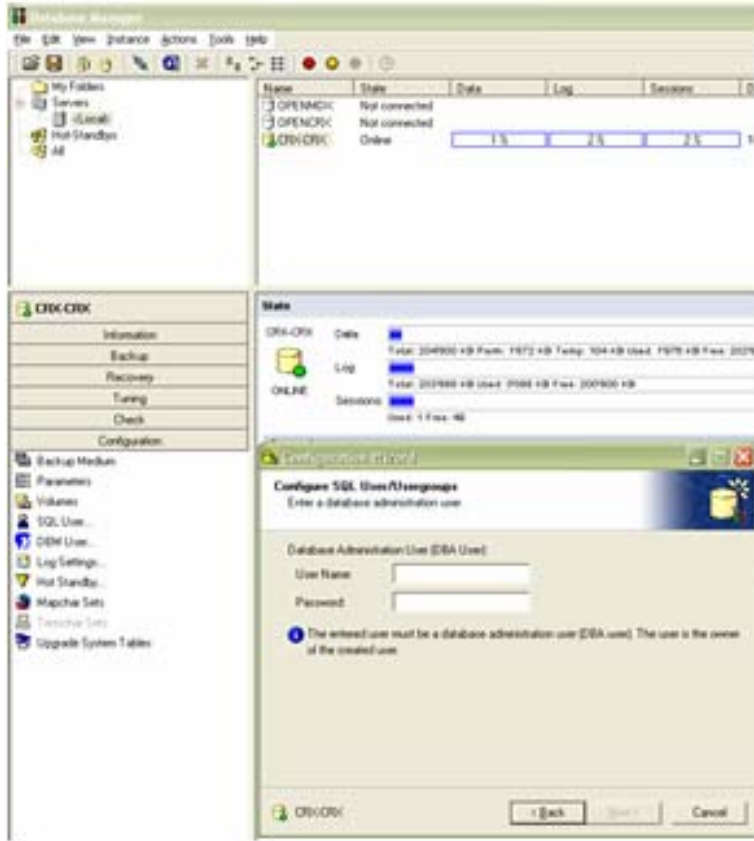
If the creation of the database instance succeeds you should finally see a dialog as shown in *Figure 4-15*.

Figure 4-15. Creation of database instance successful.



Chapter 5. Configure SQL User

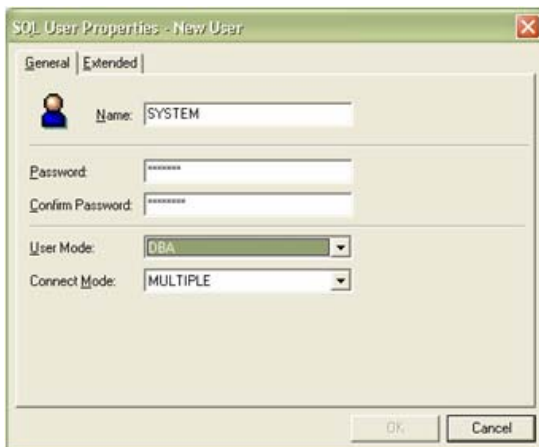
Figure 5-1. Start the SQL User management wizard.



In order that the application server is able to connect to the database you must configure a database user. After having successfully installed the database instance **crx-CRX**, you should see the instance in the database list pane of the *DBM GUI* as shown in *Figure 5-1*. Select the database instance **crx-CRX**. Then select the group **Configuration** in the lower left pane and then click the **SQL User option ...**. You must logon as database administrator. Enter **dba/dba** and proceed.

Enter the name and password of the user as shown in *Figure 5-2*. Enter the user name and password for the SQL User and set them to **SYSTEM/MANAGER**. Select the user mode **DBA**. Once you are familiar with the *openCRX* setup you can change the SQL User user and password. However, at this time we strongly recommend to use the default values.

Figure 5-2. Create an SQL User.



Chapter 6. Configure Backup



Important It is strongly recommended that you configure data and **automatic log backup**. If automatic log backup is not configured and the log files are full, *MaxDB* stops running and hangs without notice. This may result in strange behaviour of application servers (e.g. JBoss) accessing the database.

Start *DBM GUI* to activate the backup options. Select the *crx-CRX* database and click the option **Backup** in the lower left pane. Then click **Backup Wizard...** which starts the backup wizard as shown in *Figure 6-1*.

Figure 6-1. Start the backup configuration wizard.



If you do not already have performed a complete data backup you can only choose the option **Complete Data Backup**. If the option **Activate/deactivate automatic log backup** is enabled choose this option and proceed to

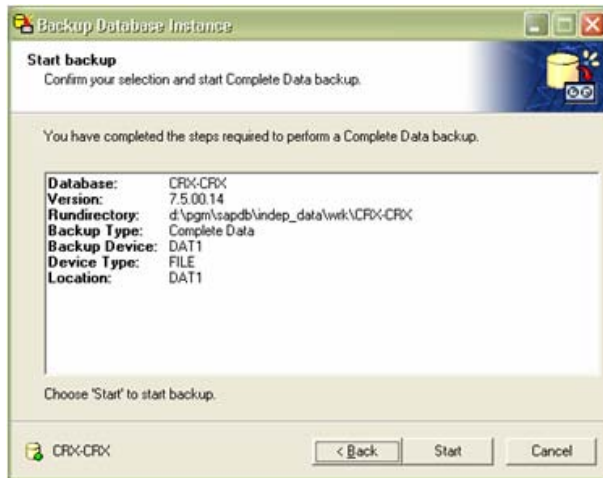
Now select an existing or create a new data backup media. The backup media should be larger than the data media you have configured earlier. This is shown in *Figure 6-2*.

Figure 6-2. Select or create a data backup media.



Now you can start the backup by clicking on Start. All your data is copied to the backup media. *Figure 6-3* shows the start backup dialog.

Figure 6-3. Start data backup.



When the data backup is complete select again the **Backup Wizard...** This time the option **Activate/dactivate automatic log backup** should be enabled. Choose this option as shown in *Figure 6-4*.

Figure 6-4. Start the backup wizard.



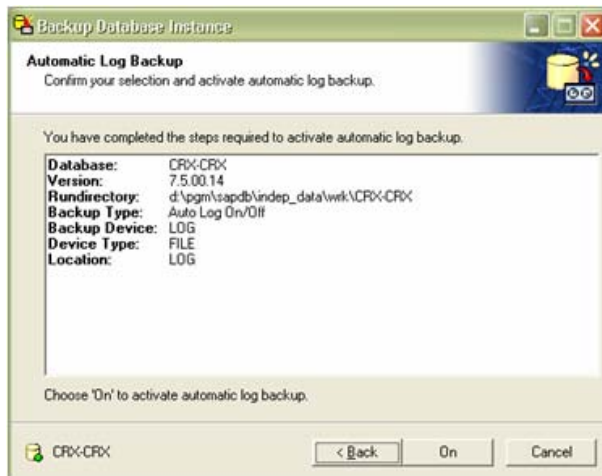
Select and existing or create a new log backup media as shown in *Figure 6-5*.

Figure 6-5. Select or create a log backup media.



The next dialog allows you to activate the automatic log backup. Select **On** as shown in *Figure 6-6*.

Figure 6-6. Activate automatic log backup.



Chapter 7. Install Database Schema

After creating the database instance and the SQL User you are ready to install the *openCRX* database schema. You have to switch now to the *SQL Studio*. Start the *SQL Studio* with the menu option **Tools -> SQL Studio** from the *DBM GUI* as shown in *Figure 7-1*.

Figure 7-1. Start SQL Studio.



SQL Studio requires you to logon to the database. Enter the user and password **SYSTEM / MANAGER** as shown in *Figure 7-2*.

Figure 7-2. Login to SQL Studio.



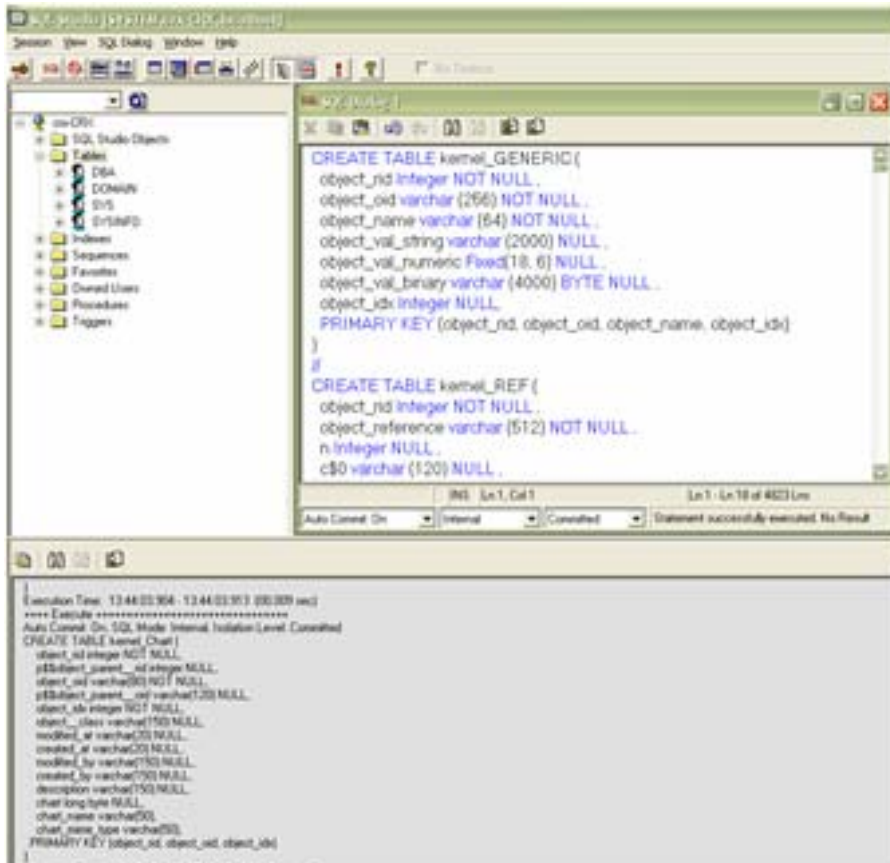
Warning All MaxDB scripts provided with *openCRX* are designed to create ASCII tables. If you created UNICODE tables you must change/adapt the scripts accordingly. In particular, you need to change/adapt those VARCHAR where you require UNICODE support. For example, you would have to change *name varchar(150) NULL* to *name varchar(150) UNICODE NULL*.

SQL Studio should automatically open an empty SQL dialog. If there is no SQL dialog open one by clicking the icon **SQL** in the toolbar. Now open the file **dbcreate-tables.sql** delivered with the *openCRX* distribution with a text editor and copy all of the content to the clipboard. Paste the clipboard to the SQL dialog and click **Execute (!)** as shown in *Figure 7-3*. This should create all tables. Now repeat the same procedure with the file **dbcreate-views.sql** and **dbcreate-indexes.sql**. All scripts should run without errors.



Warning Other scripts contained in the package must **NOT** be run at this time. They serve for other purposes such as data migration, data upgrade, etc.

Figure 7-3. Open an SQL Dialog and run openCRX schema creation scripts.



Chapter 8. Next Steps

If you have completed successfully the database installation you are ready to use the *openCRX* database. The application server installation guides explain how to connect the application server to the *openCRX* database instance.

Appendix A. Appendix

Bibliography

[01] *openCRX - the leading open source CRM solution*, [opencrx.org](http://www.opencrx.org).

@ <http://www.opencrx.org> (<http://www.opencrx.org>)

[02] *openMDX - The leading open source MDA platform*, [openmdx.org](http://www.openmdx.org).

@ <http://www.openmdx.org> (<http://www.openmdx.org>)