RIGSARKIVET

During the period 2014-2017, the Danish National Archives participated in an EU project called *E-ARK* (*European Archival Records and Knowledge Preservation*) http://www.eark-project.com/ Among other things, the project developed an *open source* extraction tool that can connect to several different *database management* systems and extract databases to a number of different preservation formats, including extracts for archival versions that live up to Executive Order no. 1007 of 20 August 2010.

Extraction tool - freely available

The purpose of this document is to disseminate the *open source Database Preservation Toolkit (DBPTK)*. The extraction tool is a program that can connect to several different *database management* systems - for example *Oracle* and *Microsoft SQL server* - and extract data content, data structures, and metadata from databases to a number of different preservation formats - including extractions for archival versions that comply with the structural provisions in Executive Order no. 1007 of 20 August 2010.

The tool is free to download from http://www.database-preservation.com/, where you can read more about the program. On the page there is also a link to *GitHub*, where the source code is freely available: https://github.com/keeps/db-preservation-toolkit. On the latter link, you can report errors and omissions to the program, even *upload* suggestions for an improved source code or get a fuller description and documentation of the program's functions.

On the following pages you can read about the more technical details of the program, so that interested parties have the opportunity to assess whether the tool can be useful. Be aware that this is an *open source* tool that may be constantly evolving, and the most recently updated documentation of functions will currently be found at https://github.com/keeps/db-preservation-toolkit/wiki/Application- usage . Also note that even if the tool is functional, it may still be necessary to carry out further preparation before or after extraction in order for the archival version to be approved according to Executive Order no. 1007 - you can also read about this on the following pages.

The Danish National Archives disclaims any liability for errors that may arise in connection with the use of the program. Please note that the National Archives cannot answer questions and provide support on the program - questions regarding the program should and can be addressed at https://github.com/keeps/db-preservation-toolkit.

Contents

1 General technical information

DBPTK is a java-based program that simply requires that Java Runtime Environment is installed. The tool can thus be used on operating systems such as Windows, Mac OS, Linux, and Solaris. The Java Runtime Environment can be downloaded at https://www.oracle.com/downloads/index.html.

A user interface for the program has not yet been created, so for now you can use the tool via a command prompt.

DBPTK consists of two modules: an import module and an export module. *DBPTK* can import and export data, data structures, and metadata from various *database management* systems and preservation formats, which can be seen in the following image:

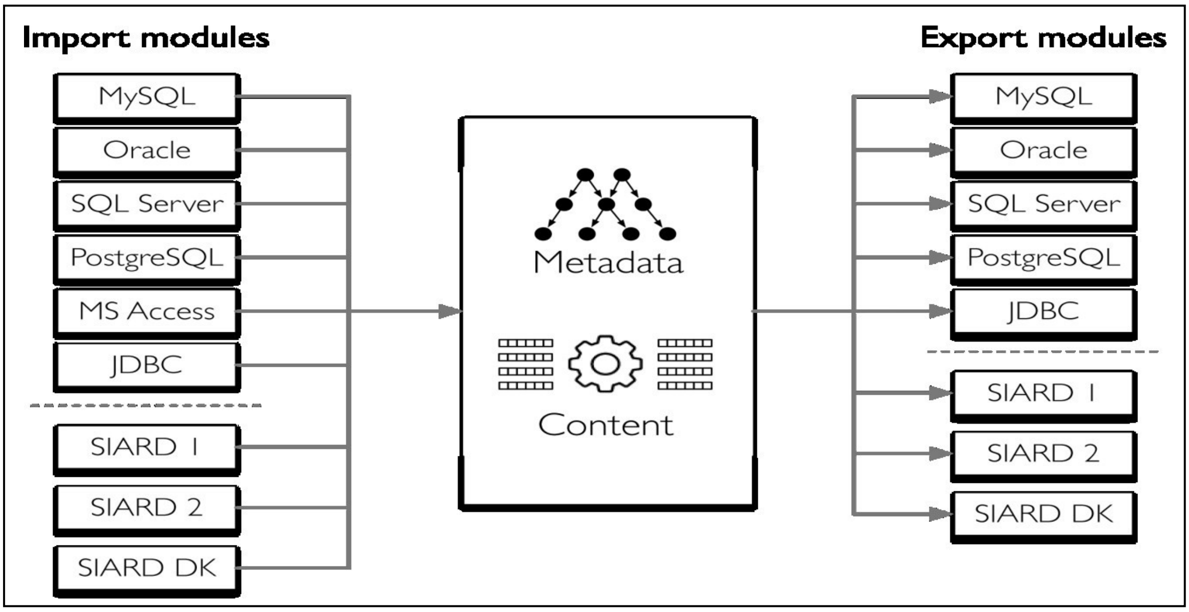


Image from KEEP SOLUTIONS[1]

It is thus possible via *DBPTK* to copy data from a running database and create an archival version in one of the three preservation formats that *DBPTK* supports.

DBPTK is *open source* and the licensing rights[2] is the GNU Lesser General Public License, version 3.

[1] *KEEP SOLUTIONS (KEEPs)* is a Portuguese company that provides *open source* solutions to the world of archives. *KEEPs* was a participant in the E-ARK project and has developed (and is still developing) DBPTK http://www.keep.pt/

[2] http://www.gnu.org/licenses/lgpl-3.0.html

2 Executive Order no. 1007 = *SIARD DK* (what should you extract to?).

SIARD is an acronym for Software Independent Archiving of Relational Databases and *“SIARD 1”* is an open Swiss conservation format, the first conservation format in the *SIARD* - “family”. The format was developed by the *Swiss Federal Archives*[3] and was included in 2008 as the official conservation format in the European long-term conservation project *PLANETS*[4] .

The Danish National Archives used *SIARD 1* as a basic structure for the preparation of Executive Order no. 1007 of 20 August 2010, “Executive Order on archival versions”. Under international framework, Executive Order no. 1007 is therefore known as *SIARD DK*. It is therefore the SIARD DK module you need in order to extract to a Danish archival version in accordance with Executive Order no. 1007.

In the *E-ARK* project, the partners, together with the *Swiss Federal Archives, have* developed the *SIARD 2* format[5] and at the same time developed *DBPTK* that can extract data to the *SIARD 2* format. Since *SIARD 1* and *SIARD DK* differ on a few points, the project ensures that *DBPTK can* also import and export from and to these modules.

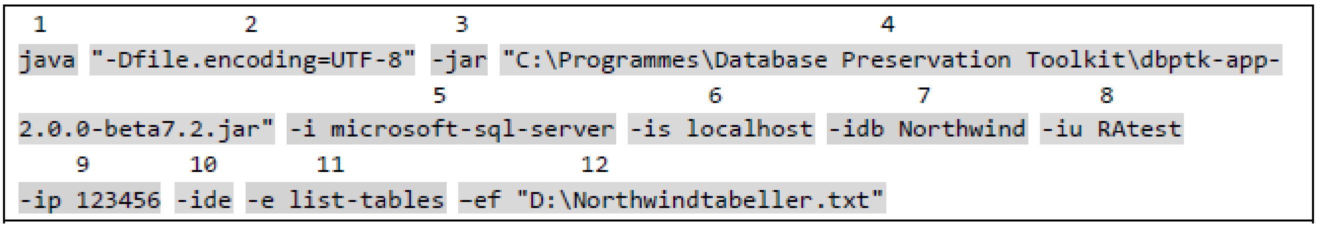
3 Parameters for the modules - example of extraction

For documentation of the parameters that can be used for the tool, you should visit https://github.com/keeps/db-preservation-toolkit/wiki/Application-usage. However, here is an example of a single extract from a test database on a *Microsoft SQL Server* for *SIARDDK*, where you have the option to choose which tables are to be included in the extract *:*

The following command is executed via a command prompt:

```
java "-Dfile.encoding=UTF-8" -jar "C: \Programmes\Database Preservation Toolkit\dbptk-app-2.0.0-beta7.2.jar" -i microsoft-sql-server -is localhost -idb Northwind -iu RAtest -ip 123456 -ide -e list-tables -ef "H:\Northwindtabeller.txt"
```

This command can be divided into the following:



```
 1              2              3                            4
java "-Dfile.encoding=UTF-8" -jar "C:\Programmes\Database Preservation Toolkit\dbptk-app-
                              5                   6             7            8
2.0.0-beta7.2.jar" -i microsoft-sql-server -is localhost -idb Northwind -iu RAtest
   9      10       11            12
-ip 123456 -ide -e list-tables -ef "D:\Northwindtabeller.txt"
```

1: Specifying that a java program is running.
2: Specifying character code sets. Using this parameter, as specified in the example, it always recommended.
3: Specifying that a java program is running.
4: Specifying the destination on specific java programs - the latest can be downloaded at http://www.database-preservation.com.
5: -i = *importmodule*. Here *Microsoft SQL Server*.

[3] See https://www.bar.admin.ch/bar/en/home/archiving/tools/siard-suite.html

[4] http://www.planets-project.eu/

[5] http://www.eark-project.com/resources/specificationdocs/32-specification-for-siard-format-v20

6: -is = *import-server-name*. Here is an easy solution by simply specifying *localhost.*

7: -idb = database name. Here *Northwind*.

8: -iu = *user*. Specification of which user *DBPTK* should use to access the database. *DBPTK* requires login with user and password. Here the user is called *RAtest*.

9: ip = *password*. User password.

10: -ide = *disable encryption*. Using this is recommended.

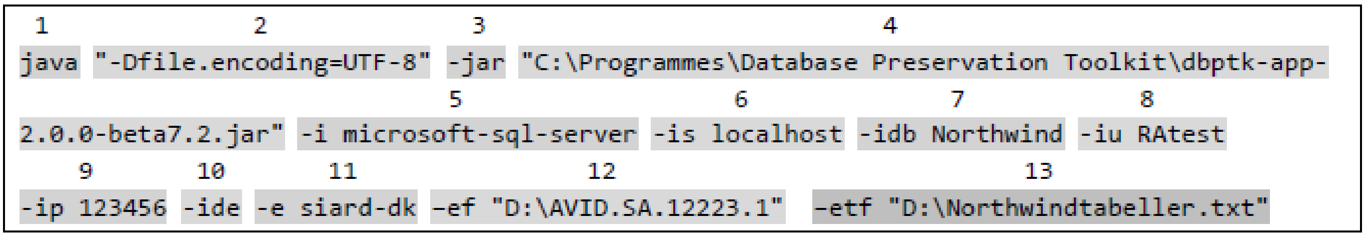
11: -e = export module. *List-tables* are specified here, which means that a text file containing the database's tables is created as the extract.

12: -ef = export folder. Here you specify the destination of the text file *DBPTK* creates. This creates a text file called "Northwindtabeller" in the root of the D drive.

The generated text file can be edited so that you can deselect tables that should not be included in the archival version. After editing, the following command can be executed:

```
java "-Dfile.encoding=UTF-8" -jar "C: \Programmes\Database Preservation Toolkit\dbptk-app-2.0.0-beta7.2.jar" -i microsoft-sql-server -is localhost -idb Northwind -iu RAtest -ip 123456 -ide -e siard-dk -ef "D:\AVID.SA.12223.1" -etf "H:\Northwindtabeller.txt"
```

This command can be divided into the following:



```
 1              2                3                            4
java "-Dfile.encoding=UTF-8" -jar "C:\Programmes\Database Preservation Toolkit\dbptk-app-
                                 5                  6              7            8
2.0.0-beta7.2.jar" -i microsoft-sql-server -is localhost -idb Northwind -iu RAtest
    9        10        11                 12                             13
-ip 123456 -ide -e siard-dk -ef "D:\AVID.SA.12223.1"  -etf "D:\Northwindtabeller.txt"
```

1: Specifying that a java program is running.

2: Specifying character code sets. It is always recommended to use this.

3: Specifying that a java program is running.

4: Specifying the destination on specific java programs - the latest can be downloaded at http://www.database-preservation.com.

5: -i = import module. Here *Microsoft SQL Server*.

6: -is = *import-server-name*. Here is an easy solution by simply specifying *localhost.*

7: -idb = database name. Here *Northwind*.

8: -iu = *user*. Specification of which user *DBPTK* should use to access the database. *DBPTK* requires login with user and password. Here the user is called *RAtest*.

9: ip = *password*. User password.

10: -ide = *disable encryption*. Using this is recommended.

11: -e = export module. Here, *"siard-dk"* is stated, which means that as an extract, an archival version is created that lives up to the structural provisions in Executive Order no. 1007 of 20 August 2010.

12: -ef = export folder. Here, in the example, you specify that *DBPTK* must create an archival version with archival version *ID* 12223, and place this in the root of the D-drive.

13: -etf = *export-table-filter*. Here you specify the destination of the text file that *DBPTK* should use as the basis for selecting which tables to extract.

4 What can DBTPK not do?

It is important to emphasise that *DBPTK* only takes the information that is in the database from which one wishes to extract data. When delivering IT systems, information outside the database itself may occur:

Table and column descriptions

System documentation. If the contents of tables and columns are described outside the database, then these will of course be missing in the created archival version. Since table and column descriptions are mandatory, cf. the Executive Order's 6.C, Figure 6.3, *DBPTK,* when creating a table index file *tableIndex.xml*, will place a dummy text in the *description* element that reads *"Description should be set manually"* for the tables and columns, where metadata does not exist.

Multiple databases

Another example where *DBPTK* cannot create a finished archival version is if a system consists of multiple databases. Here, some processing must be done either before or after extraction. *DBPTK* is not designed to import from multiple databases, and assumes that only one database should be extracted.

References/codes

If there are relationships that are not marked in the *database management* system via *constraints*, or there are coded values whose translation is not found in the database, then these will be missing in the created archival version.

Documents

If a database contains documents, embedded as BLOBs[6], DBPTK will correctly extract documents in the correct folder structure and create a document index, docIndex.xml. *However, DBPTK* extracts the documents as they are embedded with a binary stream and gives the extracted document extension *.bin*. Since Executive Order no. 1007 sets requirements for the preservation format for documents, there is therefore usually a file recognition and conversion task that must be completed before the archival version meets the document requirements specified in Executive Order no. 1007, the provisions of 4.G and 5.E.

If the documents are in external silos with references in the database, then there will also be work to place documents in the correct folder structure, convert to preservation format for documents and make sure that the references in the extract to table data comply with the provisions for document reference (dokID).

[6] https://en.wikipedia.org/wiki/Binary_large_object