Migrating to Cloudscape™ Version 5

July 8, 2002

Before using this information and the product it supports, be sure to read the information in Notices.

First Edition (August 2002)

www.ibm.com.

Table of Contents

Migrating to Cloudscape Version 5

Beginning in Cloudscape Version 5.0, some commonly used features and elements, including jar file names, the JDBC connection URL, and property names, have changed, to integrate Cloudscape more closely with the DB2 product line.

Because these changes affect pre-existing databases and applications, you must migrate Cloudscape 4.x database systems; you cannot use the automatic upgrade feature. Pre-4.0 databases need to be upgraded to 4.x, then migrated, to move them to Version 5.x.

NOTE: In this document, the letter *x* is used to include any minor or maintenance release that falls between a major release. For example, *5.x* in the sentence above refers to any Cloudscape release that begins with the major release number 5.

Upgrading from a Previous Version of Cloudscape

If you are using a pre-4.0 version of Cloudscape, you should upgrade to 4.x before attempting to migrate the database and application. This will result in fewer variables.

To get Cloudscape to upgrade a database from 3.x to 4.x, first back up the 3.x database. Next, simply connect to the database using the 4.x jar files and append the *upgrade=true* property to the connection URL. For example:

```
    Connection c =
DriverManager.getConnection("jdbc:cloudscape:myDB;upgrade=true"
);
```

NOTE: You should always back up the database that you are upgrading *from*, so you have an archival version in case you need to revert to it.

Verify Your Jar Files

To ensure you are running with the 4.x jar files, use the *sysinfo* utility:

```
java COM.cloudscape.tools.sysinfo
```

Options for Migrating Your Database

Use one of the following options to migrate your 4.x database to Cloudscape 5.x:

Your Own Program

If you have a program that creates your 4.0 database objects (tables, triggers, etc.), you can run this program using Cloudscape 5.0 or higher, to create a 5.x database.

Similarly, if you have a program that recreates your data, you can run it against your newly created 5.x database.

It is also possible to export your non-binary 4.x data using Cloudscape's import/export utility, then reload it using the 5.0 or higher code. See the *Cloudscape Tools and Utilities Guide* for more information. (If you are migrating binary data, you must either recreate it programmatically or use Cloudscape's migration tool.)

After migrating your database, you may still need to change parts of your application code to incorporate the changes made to the API and other Cloudscape features. See "Upgrading Your Application" on page -5 for information on how to do this.

Cloudscape's Migration Tool

The migration tool is a command-line Java program that loads your 4.x database and extracts the DDL used to create it. The tool uses the DDL to re-create the database objects (tables, indexes, etc.) in a 5.0 or higher database. The program also copies the user data (textual and binary, including Java objects) stored in the 4.x database into the new 5.x database.

NOTE: The new 5.x database is created in addition to your 4.x database, in a separate directory. Before migrating your database, decide on a location for the new 5.x database and make sure you have enough disk space to store both versions of the database.

The rest of this document explains how to migrate your database using the Cloudscape migration tool. It also provides information about the features and elements of Cloudscape that have changed beginning in 5.0, to help you identify any parts of your application that you need to update to work with those changes.

- Using the Cloudscape Migration Tool
- Upgrading Your Application
- What's Changed in Cloudscape 5.0?

Using the Cloudscape Migration Tool

This part of the document explains how the migration tool works and then gives you instructions on how to use the tool to migrate your database.

The migration tool has four phases:

1. First, it extracts the SQL for creating the existing 4.x database.

   This SQL is stored in a file where you run the utility and, if you want, can be used by the ij utility to recreate your 4.x schema.
2. Second, it creates the preliminary schema objects in the 5.x-level database. These are schema objects (userschemas, embedded jar files, aliases, aggregates, tables) that need to be in place before data can be loaded.
3. Third, the tool moves your data from the 4.x database to the 5.x database.

   This step is the longest in the migration process. It is highly dependent on the type of data being moved. Primitive types are fastest, Java objects the slowest.
4. Finally, the migration tool creates secondary schema objects in the 5.0-level database. (Secondary schema objects include primary, unique, and foreign keys, indexes, checks, views, statements, and triggers.)

   These objects either depend on the data or slow down the data transfer if they are created before the data is moved. It is more efficient, for example, to create primary keys after the data has been inserted rather than before.

Follow the steps below to migrate your pre-5.0 database to Cloudscape Version 5.x:

1. Upgrade your database to version 4.x.

   This is done by first backing up your pre-4.0 database, then connecting to the 3.x database with the 4.x jar files in your class path. You explicitly request an upgrade with the database connection URL attribute *upgrade=true*. For example:

```
jdbc:cloudscape:toursDB;upgrade=true
```

See the *Cloudscape Developer's Guide* for more information.

2 Put the following in your class path:

- Cloudscape 4.x jar files
- Cloudscape 5.x jar files
- the migration tool jar files: *migrationTools.jar, migrateFrom40.jar*

If your database includes any java objects, be sure to include those jar files in your class path, as well.

NOTE: For the migration tool to work, the migration tool jar files *must* appear after the Cloudscape 4.x jar files in your class path.

For example:

```
set CLASSPATH=%DB2J_INSTALL%\lib\db2j.jar;
%DB2J_INSTALL%\lib\db2jtools.jar;
%DB2J_INSTALL%\lib\db2jcview.jar;
%DB2J_INSTALL%\lib\jh.jar;
%CLOUDSCAPE_INSTALL%\lib\cloudscape.jar;
%CLOUDSCAPE_INSTALL%\lib\cloudutil.jar;
%CLOUDSCAPE_INSTALL%\lib\cloudview.jar;
%CLOUDSCAPE_INSTALL%\lib\jh.jar;
%CLOUDSCAPE_INSTALL%\migration\migrationTools.jar;
%CLOUDSCAPE_INSTALL%\migration\migrateFrom40.jar;%CLASSPATH%
```

3 The usage for the tool is:

```
java [OPTIONS]
COM.cloudscape.tools.migration.MigrateFrom40 <sourceDBname>
```

The source name is the db name only, *not* the URL. For example, *myDB* or *c:\temp\myDB*; not *jdbc:cloudscape:myDB.*

Both the 4.0 and 5.0 databases are booted in embedded mode.

Options include:
-*DappendLogs=true*, to keep from overwriting the output logs.

-*DsqlLogName=*<*name*>, to specify a file to write the SQL to recreate the 4.0 level DB.
This defaults to <*sourceDBname*>.*sql*

-*DnewDBname=*<*name*>, to specify the name of the new database.
This defaults to <*sourceDBname*>.<*newversion*>

If you need further assistance migrating you database, go to *http://www.ibm.com/software/data/cloudscape/support/*.

Upgrading Your Application

After you've migrated the database to 5.x, you may also need to migrate your application. (Most applications, especially those that do not make use of Cloudscape's ability to use Java objects as column types, will need little or no changing.)

If your application does need to be changed, here are some steps to take to get you started:

1. First, replace the 4.x jar files with the 5.x jar files in your class path, and recompile your program.

NOTE: Be sure your class path is correct. (You can verify your class path by running *sysinfo*.)

2. Change the driver name and JDBC URL.
   The new JDBC driver is called *com.ibm.db2j.jdbc.DB2jDriver.*
   The new URL for connecting to a database is: *jdbc:db2j:<yourDBname>*
3. Review the section "What's Changed in Cloudscape 5.0?" on page -5.
   Use this list to identify changes you need to make to your code.
   We suggest that you search through your code for *cloud*, as a quick way to locate some of the places you will need to make changes.
4. Some Cloudscape APIs have changed, others have disappeared. Use any compilation errors you get to help find the parts of your code that depend on these APIs. Consult the Cloudscape 5.x documentation for specific information on the classes the compiler is complaining about.

What's Changed in Cloudscape 5.0?

Use the information below as a reference to commonly used features and elements that have changed in Cloudscape 5.0 and later versions.

- JDBC driver name
- JDBC URL
- authentication scheme
- autocommit and JDBC
- CLASS keyword
- Cloudconnector
- Cloudscape API

- Cloudscape properties
- Cloudsync
- decimal types
- getting the current JDBC connection
- INSTALL environment variable
- jar files
- LONGINT changed to BIGINT
- RmiJdbc
- SERIALIZE keyword
- static field and method access uses :: construct
- system-level class loading
- user-defined aggregates

JDBC driver name

The new JDBC driver is called *com.ibm.db2j.jdbc.DB2jDriver.* This is loaded in the same way as any other driver:

```
Class.forName("com.ibm.db2j.jdbc.DB2jDriver");
```

JDBC URL

The URL for connecting to a database is: *jdbc:db2j:<yourDBname>*
This URL has been registered with Sun.

authentication scheme

The class *com.ibm.db2j.authentication.Interface.AuthenticationScheme* has been renamed. The new name of the class is c*om.ibm.db2j.authentication.UserAuthenticator.*

The class *com.ibm.db2j.authentication.Interface.AuthenticationException* has been replaced with *java.sql.SQLException.*

The *SQLState* for a user authentication error is now the standard 08004, not XJ006.

autocommit and JDBC

Use *setAutoCommit()* method, the standard way to set autocommit.

CLASS keyword

If your application contains code which references the CLASS keyword:

```
CALL (CLASS MyClass).staticMethod('foo')
```

you must change your code to use the following syntax:

```
CALL MyClass::staticMethod('foo')
```

This is more readable and conforms more closely to the existing SQL standard.

Cloudconnector

Cloudconnector is no longer available, beginning in version 5.0. IBM Websphere is available as an alternate solution.

Cloudscape API

You must change the names of Cloudscape packages and classes from *COM.cloudscape.** to *com.ibm.db2j.**

See "Appendix A" in the *Cloudscape Reference Manual* for the changes made in Version 5.0 to specific classes.

Cloudscape properties

If your application was making use of system properties (such as *cloudscape.storage.pageSize*), you must modify them. Generally, you must change the name from *cloudscape.xxx.yyy* to *db2j.xxx.yyy.* Refer to *Tuning Cloudscape* for information about specific properties.

Properties file

db2j.properties

If your application was using the *cloudscape.properties* file to set these properties, you will need to rename this file d*b2j.properties*. You must also change property names from *cloudscape.<propertyname>* to *db2j.<propertyname>*.

service.properties

This is an internal file used by Cloudscape code to configure itself at boot-time. Changes made to this file prevent a 4.x database from being booted by 5.x code and vice versa. *Do not ever make changes to this file.*

Cloudsync

Cloudsync is no longer available, beginning in version 5.0. Cloudscape will have a new synchronization solution available soon.

decimal types

The format for decimal types has changed. To be more in line with DB2 and realistic datasets, the upper limit for scale and precision for DECIMAL types has been changed to 255. See the user documentation for information about DECIMAL types.

getting the current JDBC connection

Use the standard URL, *jdbc:default:connection*, for the current connection.

INSTALL environment variable

DB2J_INSTALL

jar files

db2j.jar

db2jtools.jar

db2jcview.jar

LOG file

db2j.log

LONGINT changed to BIGINT

To conform to SQL standard, in Cloudscape 5.0 and higher, the datatype BIGINT replaces LONGINT.

RmiJdbc

RmiJdbc is no longer shipped with Cloudscape, beginning in version 5.0. It is being replaced with a new solution, available in version 5.1. As in previous versions, you can also embed Cloudscape in another server framework.

SERIALIZE keyword

The SERIALIZE keyword is no longer needed or recognized in Cloudscape 5.0 and higher. The parser can tell from the structure of the statement that you are using a Java object.

If your application uses a construct such as:

```
stmt.execute("CREATE TABLE T1 (idx INT NOT NULL, person
(SERIALIZE package.MyPerson))");
```

it needs to be changed to:

```
stmt.execute("CREATE TABLE T1 (idx INT NOT NULL, person
package.MyPerson)");
```

static field and method access uses :: construct

To match SQLJ part 2 static field and method access, use:

```
<java class name or class alias>::<method>(...)
```

and

```
<java class name or class alias>::<field>
```

In Cloudscape 5.x, static fields and methods now use the :: construct. In Cloudscape 4.x, the correct syntax for accessing fields on objects stored in the database was the -> construct. Referencing method calls used the . syntax. No distinction was made between instance fields and methods and static fields and methods (those shared by the entire class).

Table 0-1, “Static Field and Method Access in 4.x and 5.x” shows the differences between versions.

Table 0-1 Static Field and Method Access in 4.x and 5.x

Type	4.x	5.x
Instance Method	.	.
Static Method	.	::
Instance Field	->	->
Static Field	->	::

system-level class loading

In Cloudscape 5.1 and later versions, system-level class loading is no longer supported. For information on database-level class loading, see *Tuning Cloudscape*.

user-defined aggregates

User-defined aggregates are not available in Cloudscape 5.0 and Cloudscape 5.1. They will be reimplemented in a later version of Cloudscape.

Notices

This information was developed for products and services offered in the U.S.A. IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY

OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
J46A/G4
555 Bailey Avenue,
San Jose, CA 95141-1003
U.S.A

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

Trademarks

Cloudscape™

DB2™

IBM™

Websphere™

The following terms are trademarks or registered trademarks of other companies:

Microsoft™, Windows™, and Windows 2000™ are trademarks or registered trademarks of Microsoft Corporation.

Java™ or all Java-based trademarks and logos, and Solaris™ are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

UNIX™ is a registered trademark in the United States, other countries or both and is licensed exclusively through X/Open Company Limited.

Other company, product, or service names may be trademarks or service marks of others.