

IBM Cloudscape on the IBM iSeries platform

*Installation, invocation,
development tools, and more*

*IBM eServer Solution Enablement
September 2005*



Table of contents

Abstract	3
Introduction	4
Prerequisites.....	4
Configuring Cloudscape graphically	4
Configuring and starting the iSeries VNC server job.....	5
Installing Cloudscape	11
Starting the Cloudscape network server	21
Entering a QSH environment using the STRQSH command.....	21
Exploring sample Java applications	22
Creating a table and inserting rows into the database	24
Query the database	25
Using the Cloudscape interactive tools	26
Using the Derby interactive JDBC scripting tool	26
Invoking ij options	26
Creating a schema, table, and data with ij.....	27
Accessing and deploying a Cloudscape database using RAD	28
Using an existing schema	28
Using the interactive SQL Scrapbook.....	32
Using RAD to create a new schema and table.....	34
Summary	48
Resources	49
Trademarks and special notices	50

Abstract

IBM® Cloudscape™ is a pure, open source-based Java™ relational database management system that can be embedded in Java programs and used for online transaction processing (OLTP). A platform-independent, small-footprint (2 MB) database, Cloudscape Version 10.0 and Version 10.1 integrate tightly with any Java solution.

This white paper will guide the reader through the process of setting up an IBM eServer™ iSeries™ system to allow for graphical installation of Cloudscape.

The reader will also learn how to start a Cloudscape database using network server mode and will be provided with some Java samples that can be run against the Cloudscape server.

Introduction

Cloudscape is a fully relational Java database that IBM has released as open source to the Apache Software Foundation. The Apache consortium has established Cloudscape as an incubator project under the name *Derby*. Cloudscape is a database engine written completely in Java and will run in any certified Java Virtual Machine (JVM™). (Refer to the "Resources" page for a link to the Cloudscape Web site that provides a list of the specific releases for various platforms and Java environments in which Cloudscape V10.0 and V10.1 has been tested.)

Cloudscape can be deployed in a number of ways, including the following:

- Embedded in a single-user Java application. Cloudscape can be practically invisible to the user, because it requires no administration and runs in the same JVM as the application.
- Embedded in a multi-user application such as a Web server, an application server, or a shared development environment
- Embedded in a server framework through either the network server or a user-defined server

Prerequisites

The prerequisites for installing Cloudscape are listed below. (Refer to the "Resources" section for download and other information.)

- Cloudscape
 - Version 10.0
 - Version 10.1
- iSeries Tools for Developers (IBM LPP: 5799-PTL)
- IBM i5/OS® V5R3
- VNC (Virtual Network Computing) viewer on desktop downloadable from the RealVNC™ Web site located in the "Resources" section of this paper.
- IBM Rational® Software Development Platform Version 6

Configuring Cloudscape graphically

The Cloudscape installer/uninstaller has both an interactive and silent mode. This paper will use the interactive graphical user interface (GUI) wizard interface. Because i5/OS does not support native GUI devices, use the VNC support that comes with the iSeries Tools for Developers product included in the i5/OS, licensed program product 5799-PTL. (**Note:** To view the installed licensed programs on the target iSeries server, at the command line, type in **GO LICPGM** and use option 10.)

The VNC client viewer on the Microsoft® Windows® or Linux™ workstation is also needed, which can be downloaded at the Web site mentioned in the “Prerequisites” section.

Configuring and starting the iSeries VNC server job

The VNC server must be started and running on the iSeries server to remotely stream the GUI install wizard to the PC or Linux desktop. Take these steps on the iSeries server itself.

1. Log on with a user ID that has ALLOBJ authority and is capable of creating directories and system environment variables.
2. Create the directories to store the password for the VNC server:
 - a. **MKDIR DIR('/home/somedirectory')**
 - b. **MKDIR DIR('/home/somedirectory/.vnc')**
3. Create the password file required by the VNC server. There will be a prompt to fill in a password (maximum of seven characters). On the iSeries command line type:
 - a. **QAPTL/VNCPASSWD USEHOME(*NO)
PWDFILE('/home/somedirectory/.vnc/passwd')**
 - b. Note (in Figure 1) the use of **/home/QEJBSVR/.vnc** as the directory where the **VNC passwd** file will be created and stored.
 - c. Then press **Enter**.

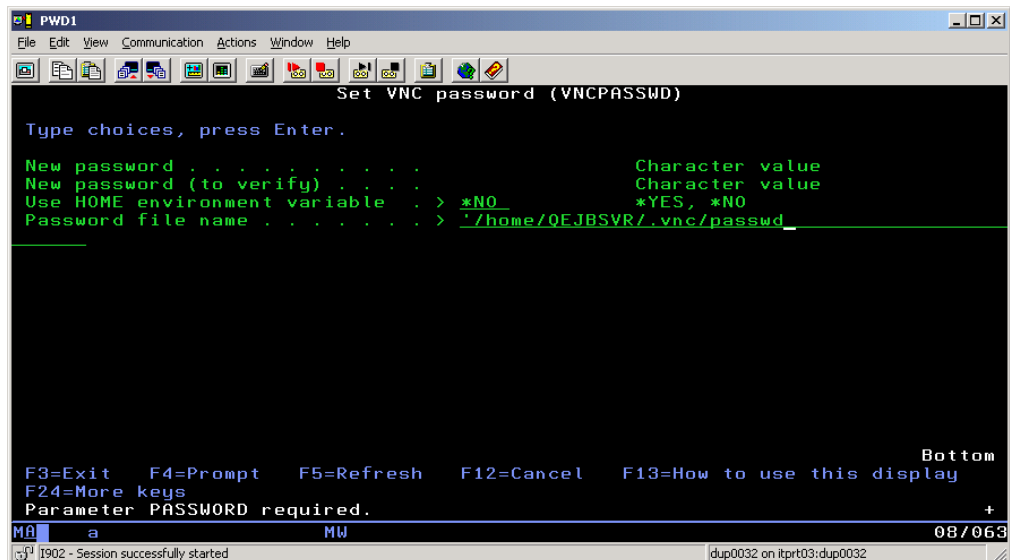


Figure 1: Set VNC password

- Assign the appropriate permissions to the `/home/QEJBSVR` directory tree. On the iSeries command line type:

STRQSH CMD('chown -R <i5user> /home/QEJBSVR') where the eServer i5 user will be the person who starts the VNC server.

- Go to the IBM Portable Application Solutions Environment (PASE) console by entering **CALL QP2TERM** from the iSeries command line. If it fails with a "Terminal session already in use" message, then the active QSHELL session will have to be ended. To end an active QSHELL session:
 - From the iSeries command line enter **QSH**.
 - Press **F3** to exit.
 - Back at the iSeries command line retry the **CALL QP2TERM** command.

- Enter **ps -ef | grep vnc** to list the already active VNC servers as follows:

```
> ps -ef | grep vnc  
  
jrush      9143      1 08-24-2005 09:36:01 -      00:22 pgm-xvnc
```

- When a line of output from the `ps -ef` command is shown (as seen in the above step), a VNC server is already up and running. To determine which VNC port is in use, utilize the **netstat *cnn** command on an iSeries command line. Page down until a local port value between 5901 and 5999 is shown. The VNC client and VNC server communicate through a port that is numbered as 5900 plus the display number (between one and 99). For example, lines like this might be shown:

Remote Address	Remote Port	Local Port	Idle Time	State
*	*	5901	001:28:11	Listen
*	*	5923	001:28:11	Listen

From this, it is possible to determine that a VNC SERVER is running and waiting for clients on the VNC display **:1** and **:23** (5901 – 5900 and 5923-5900).

8. On the iSeries command line enter **DEL '/tmp/.X11-unix/Xn** where **n** is the display number. (Use a display number that is not in the list for the netstat *cnn command above.)
9. In the SystemDefault.properties file in the home directory, or the systemwide SystemDefault.properties file in /QIBM/UserData/Java400, add this line:

os400.awt.native=true

10. Create two system environment variables. Use the iSeries command:
WRKENVVAR LEVEL(*SYS)
11. Under the **Opt** column, type **1** to add an environment variable and type **DISPLAY** under the **Name** column (Figure 2). Press **Enter**.

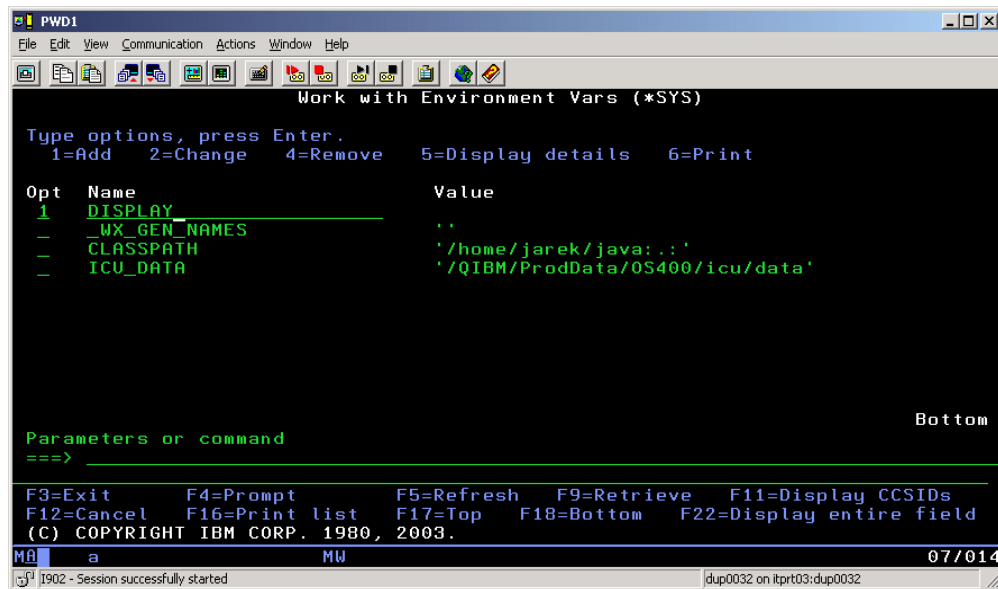


Figure 2: The Opt and Name column variables

12. In the initial value field, specify **<your hostname>:n** in the **Value** field. *Your hostname* is the TCP/IP symbolic name or IP address of the iSeries system where Cloudscape will be installed and *:n* is the VNC display number (Figure 3). Remember this number because it will be used in actually starting the VNC server program in the steps that follow. Then press **Enter**.

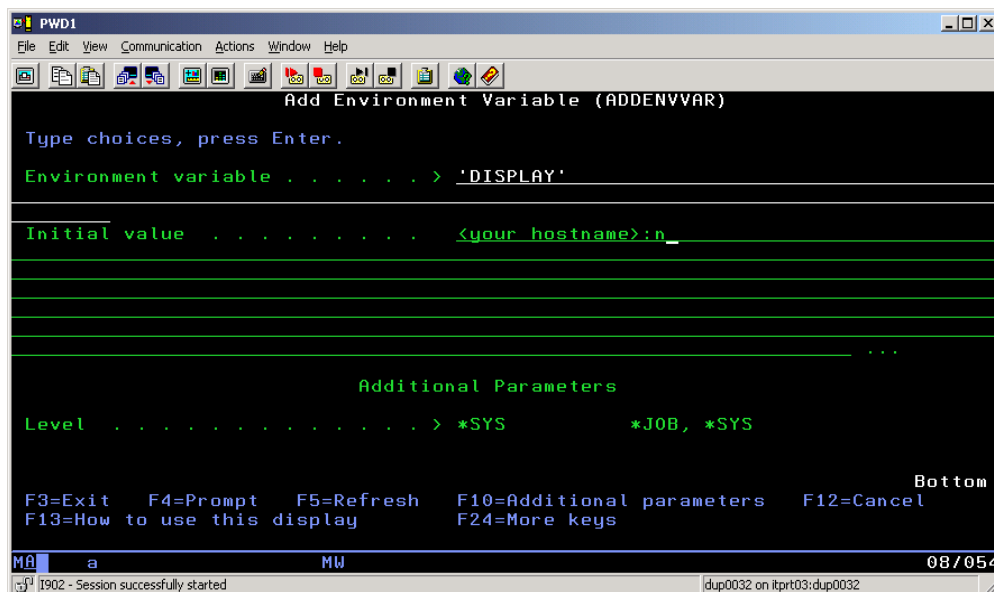


Figure 3: VNC display number

13. Type **1** in Opt to add the **XAUTHORITY** environment variable (Figure 4). Then press **Enter**.

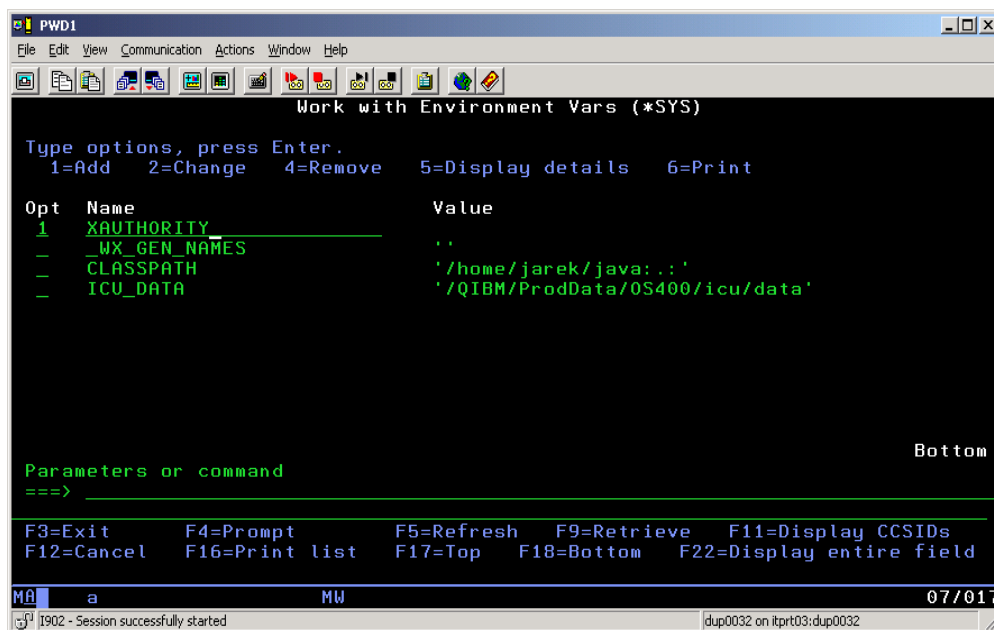


Figure 4: Add XAUTHORITY variable

14. Enter the 'XAUTHORITY' value as shown below in Figure 5. Note that /home/QEJBSVR is used in this example. Ensure that it matches the directory where the password file was created and stored. Press **Enter**.

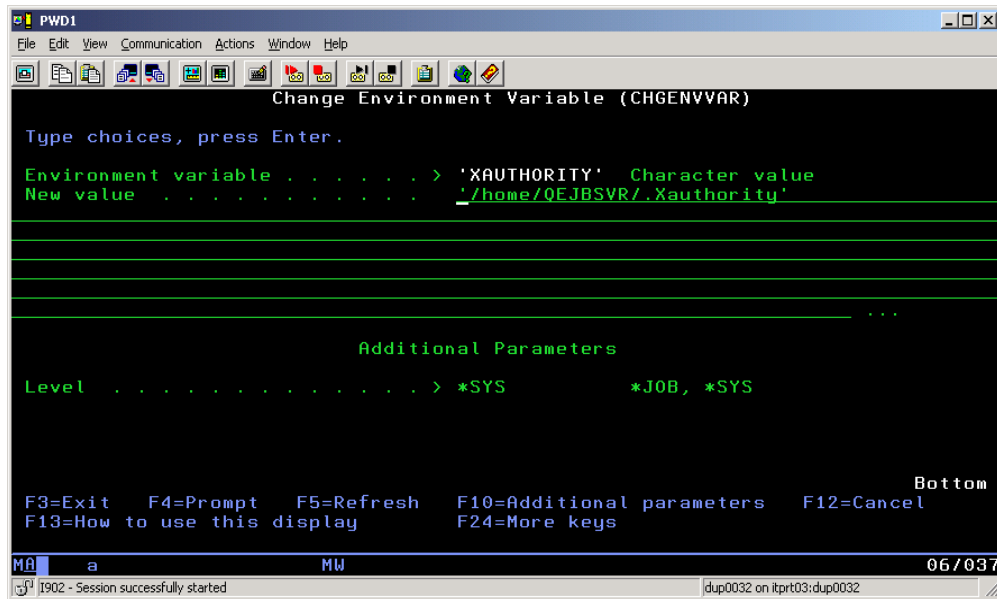


Figure 5: New value

15. Submit the VNC SERVER job to start by entering the following on the iSeries command line:

```
CALL PGM(QP2SHELL)
  PARM('/QOpenSys/QIBM/ProdData/DeveloperTools/vnc/vncserver_java' ':n')
```

... where n is the VNC display number. This also matches the number from the DISPLAY environment variable created in step 12.

Now repeat the **netstat *cnn** command to see the vncserver listening on display :nn (indicated in the red line below):

Remote Address	Remote Port	Local Port	Idle Time	State
*	*	5901	001:28:11	Listen
*	*	59nn	001:28:11	Listen
*	*	5923	001:28:11	Listen

16. On the workstation (Windows or Linux) run a VNC viewer and connect to `<your hostname>:n` in the VNC server prompt/field. *Your hostname* is the TCP/IP symbolic name or IP address of the iSeries system where Cloudscape will be installed; and *:n* is the VNC display number.
17. Enter the session password that was set up in step three. A VNC desktop window will be shown (Figure 6):

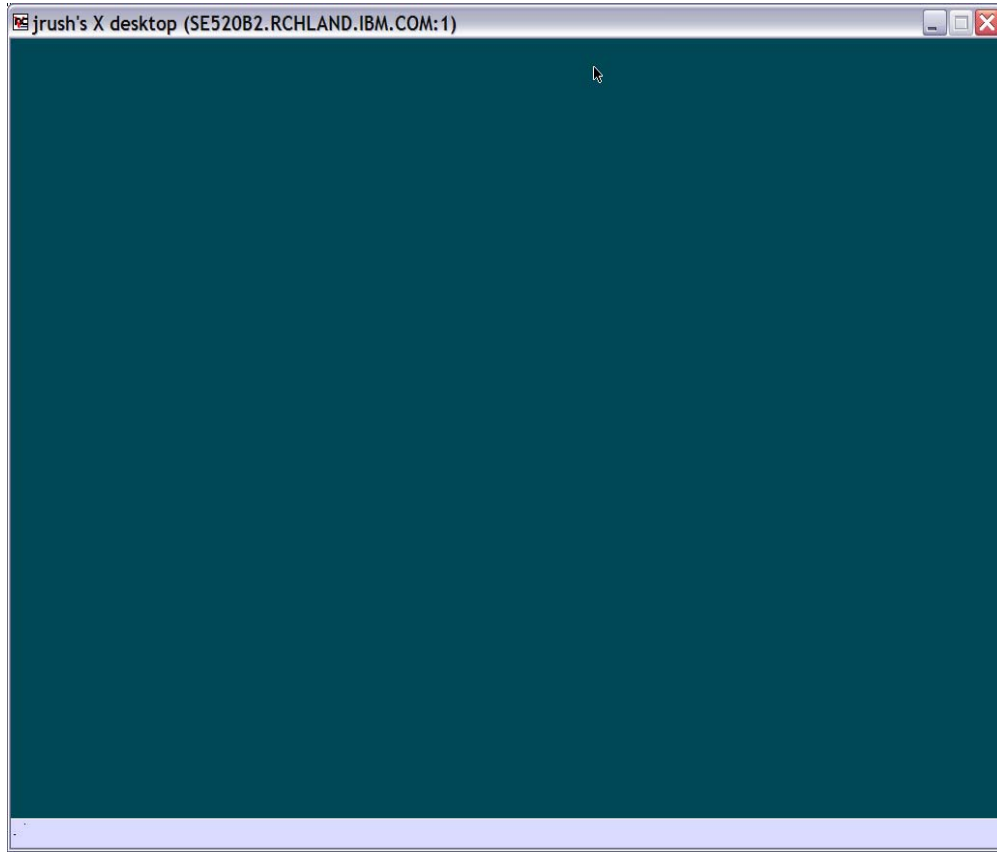


Figure 6: VNC desktop window

Installing Cloudscape

After downloading and unzipping the **10.0-IBM-Cloudscape.jar** or **10.1-IBM-Cloudscape.jar** file, begin the installation using the **strgsh** command (Figure 7).

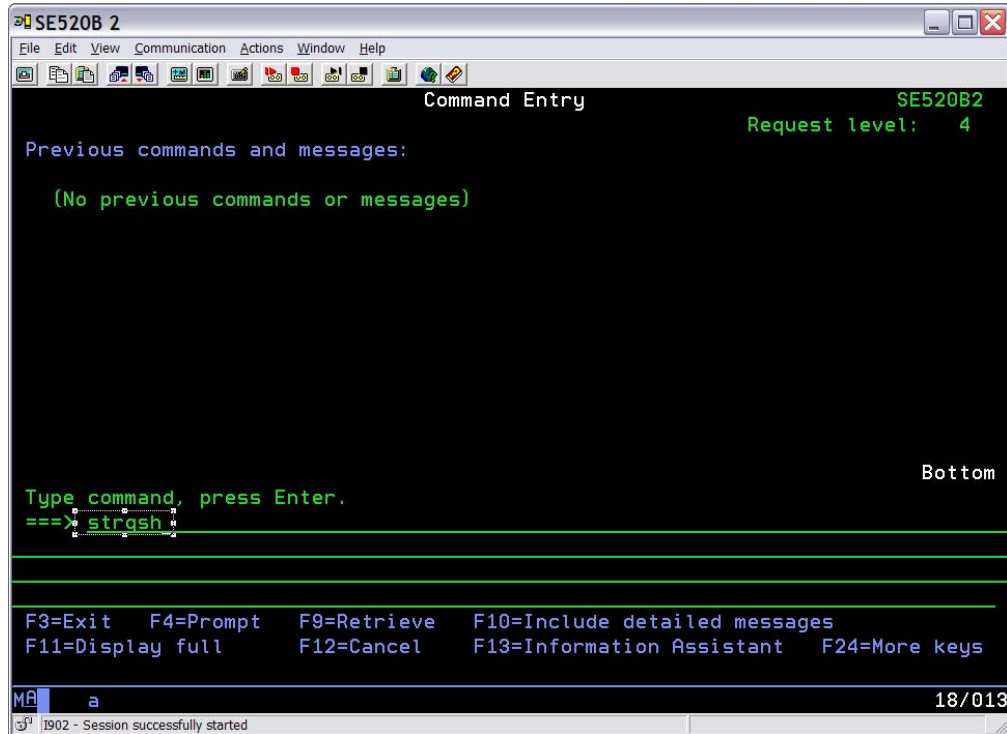


Figure 7: Cloudscape installation

1. Change to the directory where the Cloudscape installer jar is located. If using V10.1, create the Integrated File System (IFS) directory where Cloudscape is to be installed (for example, `mkdir '/IBM/Cloudscape_10.1'`). The V10.1 install will fail if this directory is not created ahead of time. The command to start the install (Figure 8) is `java -jar 10.0-IBM-Cloudscape.jar` (or `10.1-IBM-Cloudscape.jar` for Cloudscape V10.1).

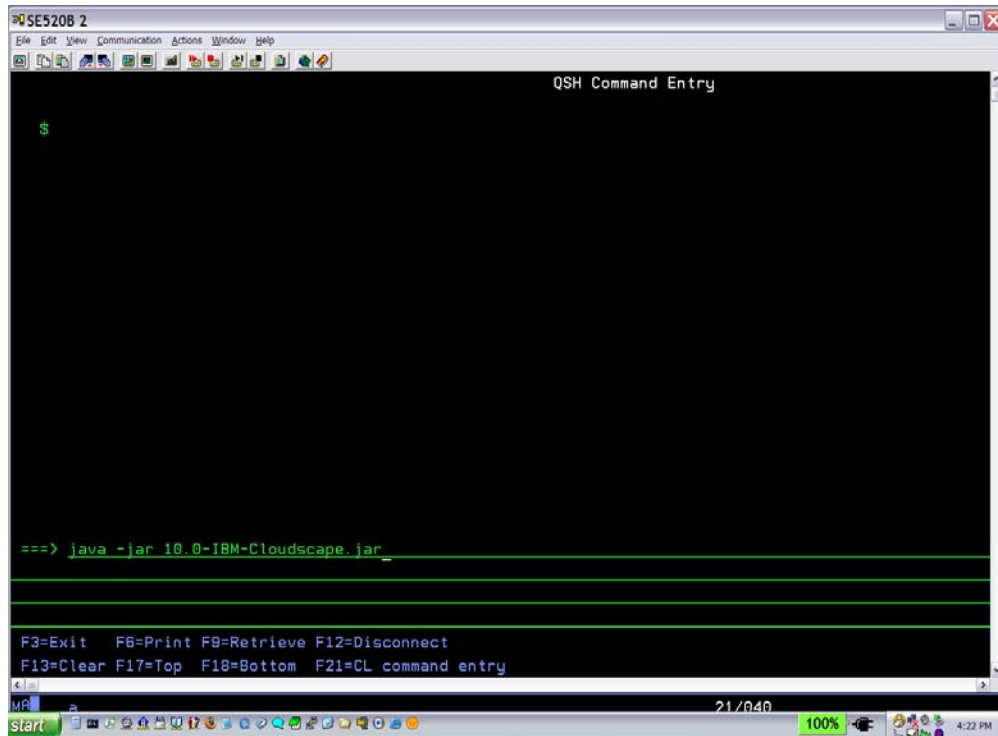


Figure 8: Start Cloudscape install

The Java installer will be invoked and the GUI wizard will be redirected to the VNC client started above.

2. Eventually (within a minute, depending on the iSeries processor) the welcome screen for the Cloudscape installer will be shown (Figure 9). Click **NEXT**.

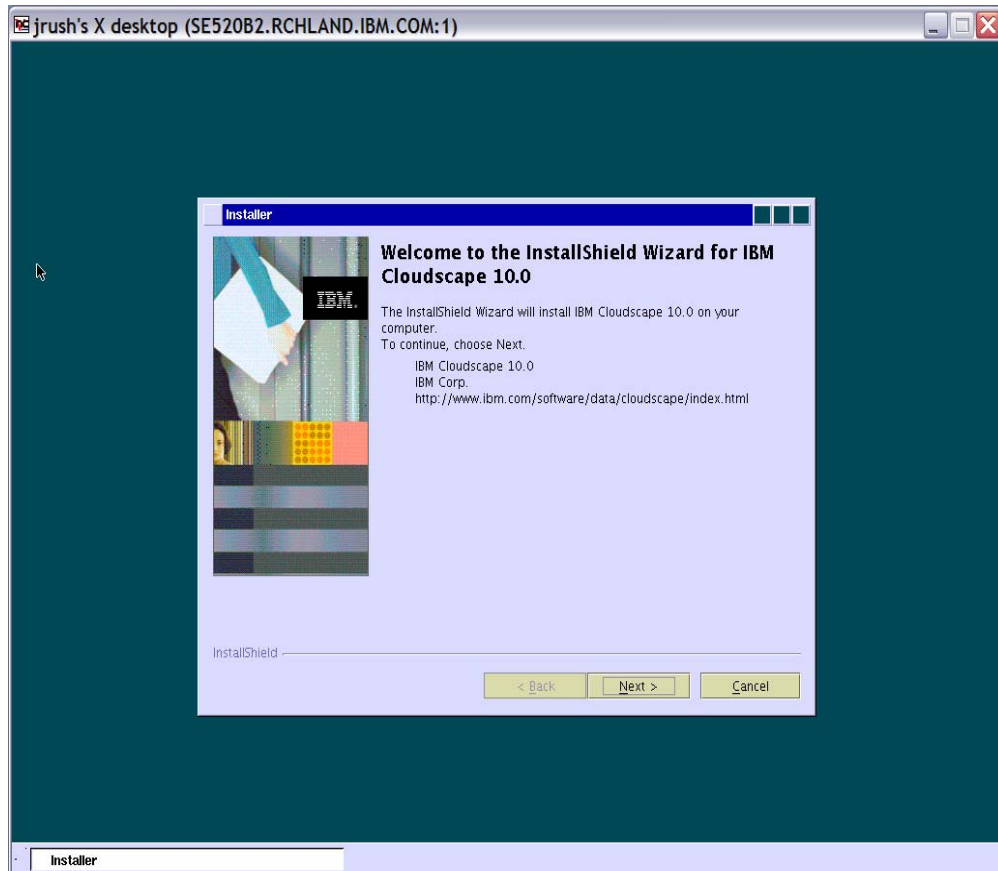


Figure 9: Cloudscape installer Welcome screen

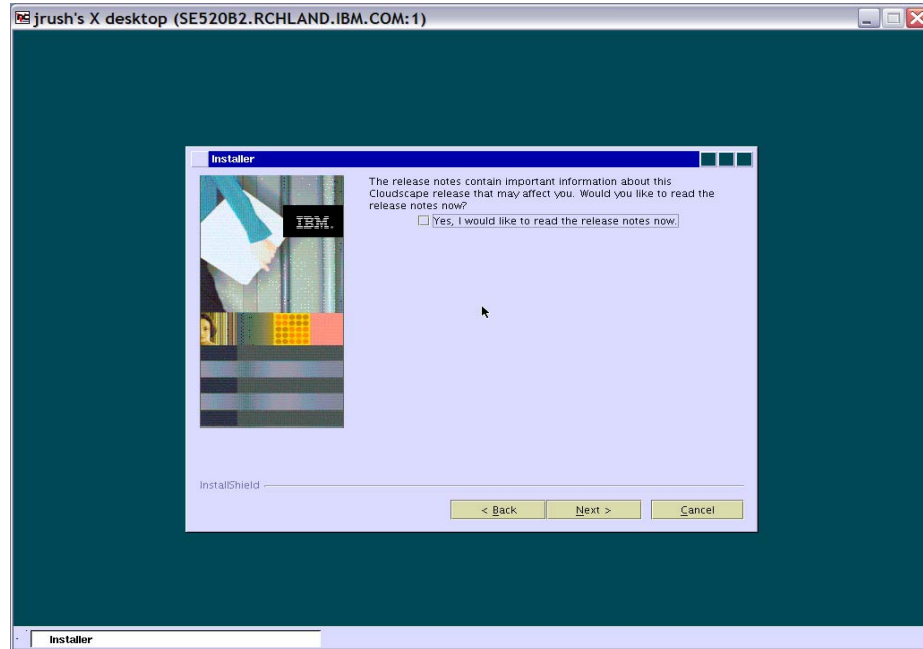


Figure 10: Release notes screen

3. On the release notes screen (Figure 10), just click **NEXT** to ready the notes. Otherwise, uncheck "Yes, I would like to read the release notes now" and then click **NEXT** to reach the license agreement (Figure 11).

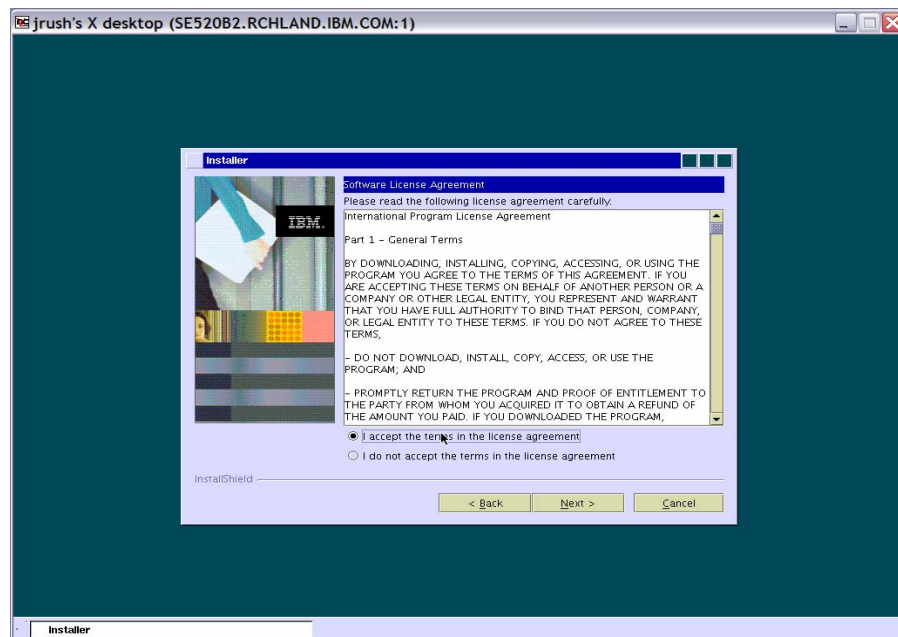


Figure 11: License agreement screen

4. Read the license agreement and select **I accept...** then click **Next** for the directory screen (Figure 12).

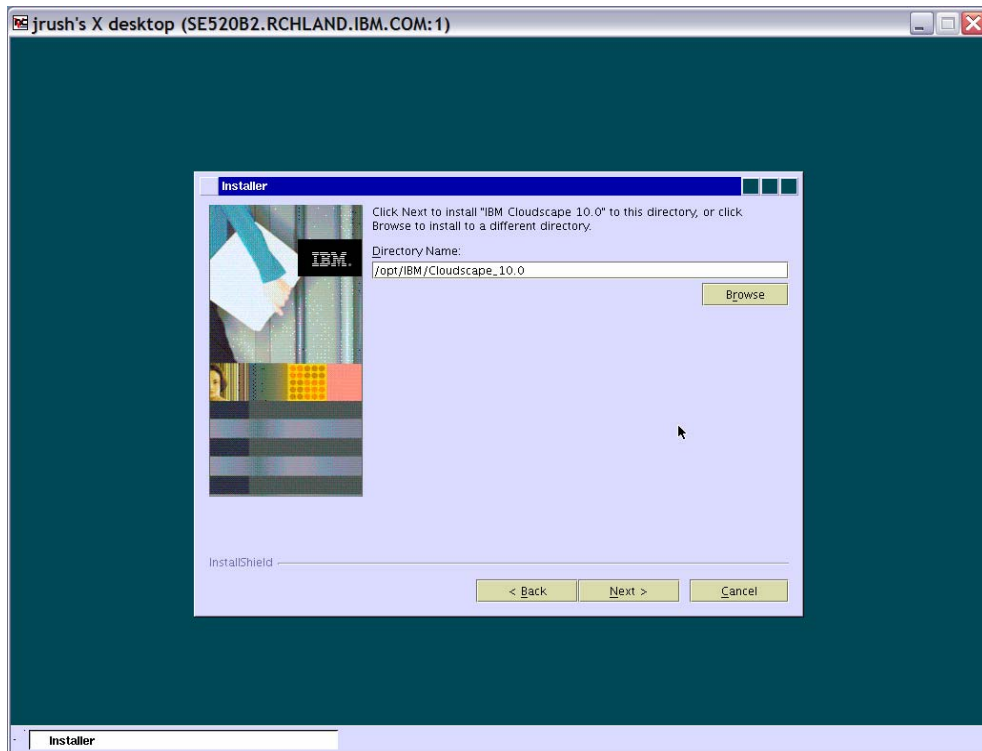


Figure 12: Choose a directory screen

5. The default directory is /IBM/Cloudscape_10.0 (/IBM/Cloudscape_10.1 for V10.1) for the install directory. If a directory is desired other than the default, enter the directory where Cloudscape is to be installed, and click **Next** (Figure 12).

6. Select the **Typical** install (Figure 13) and click **Next**.

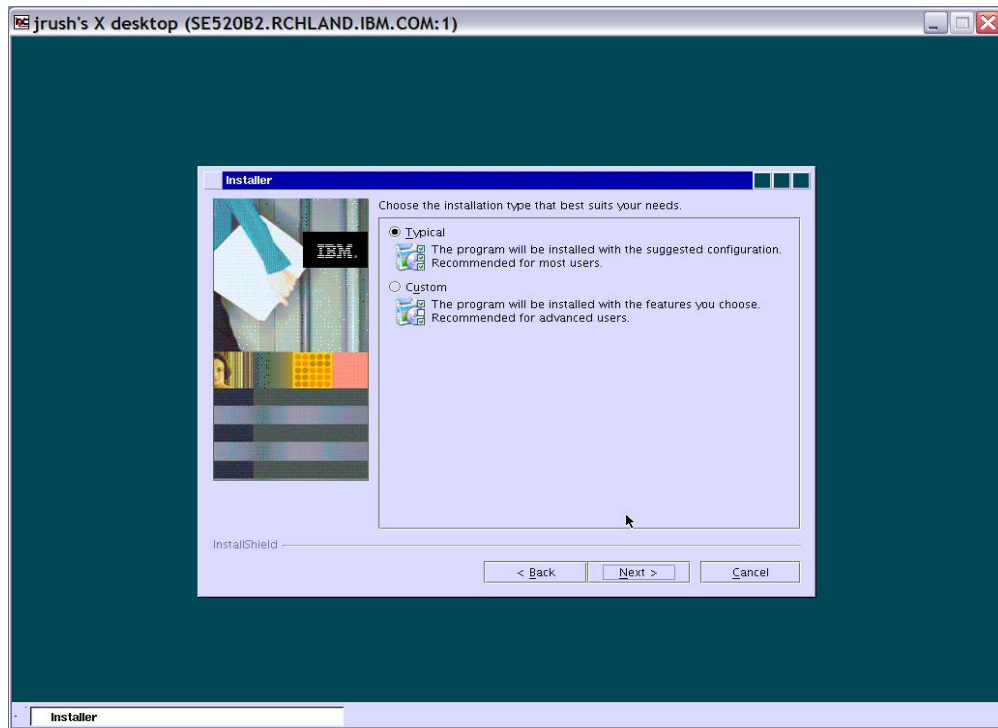


Figure 13: Choose the installation type

7. Figure 14 is the confirmation screen. Click **Next**.

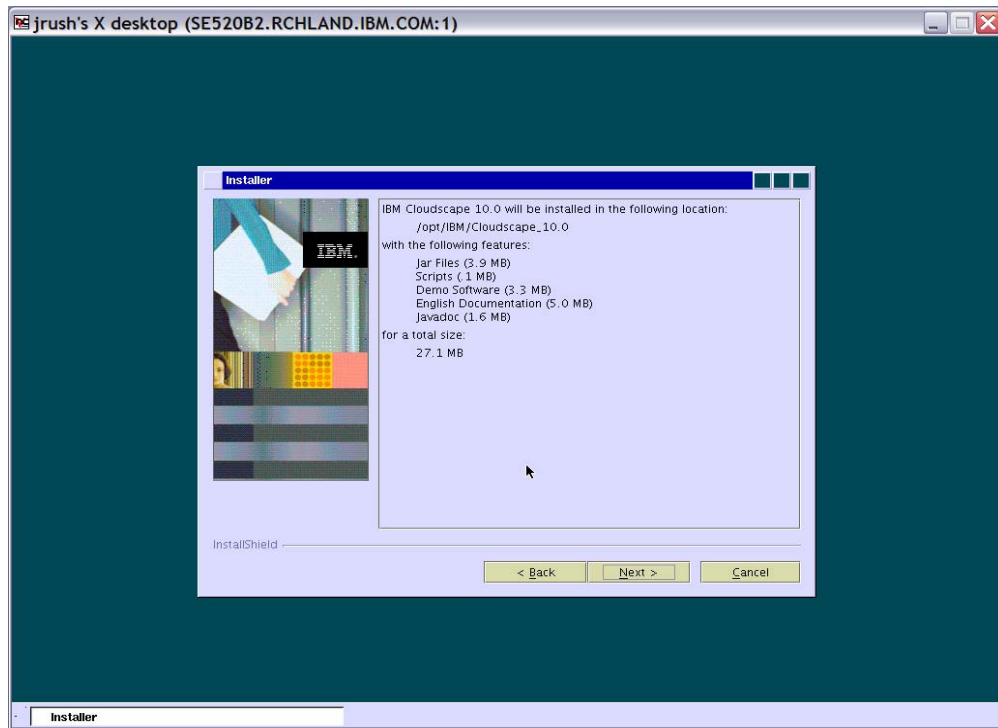


Figure 14: Confirmation screen

8. Figure 15 indicates the installation's progress.

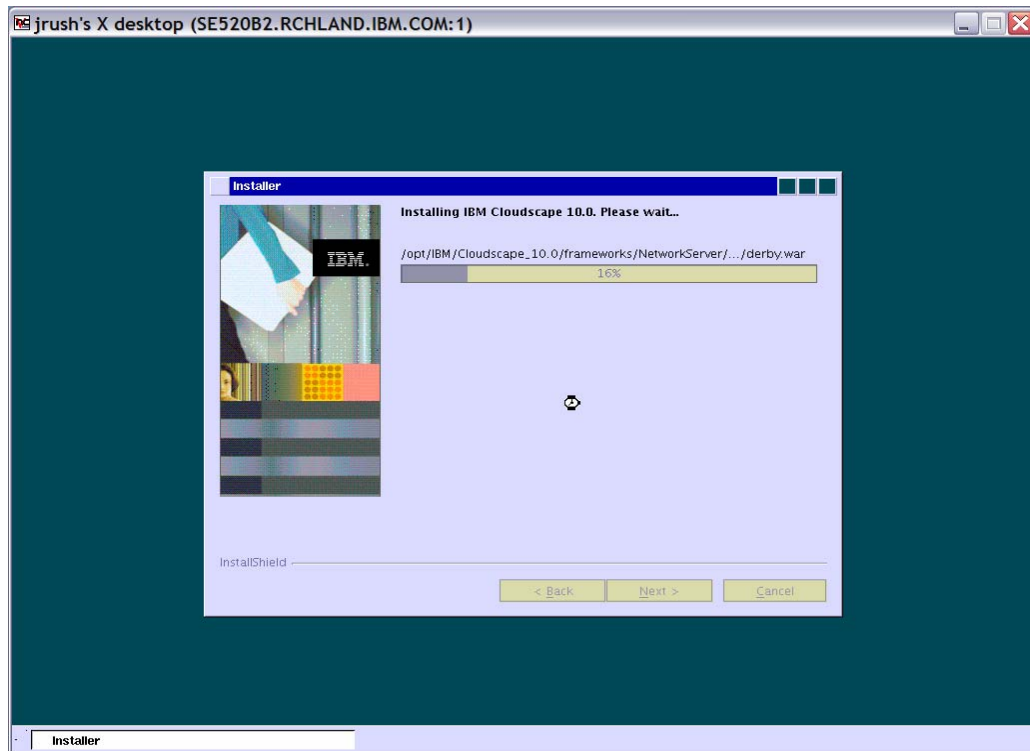


Figure 15: Installation progress

9. When prompted (Figure 16), click **Next**.

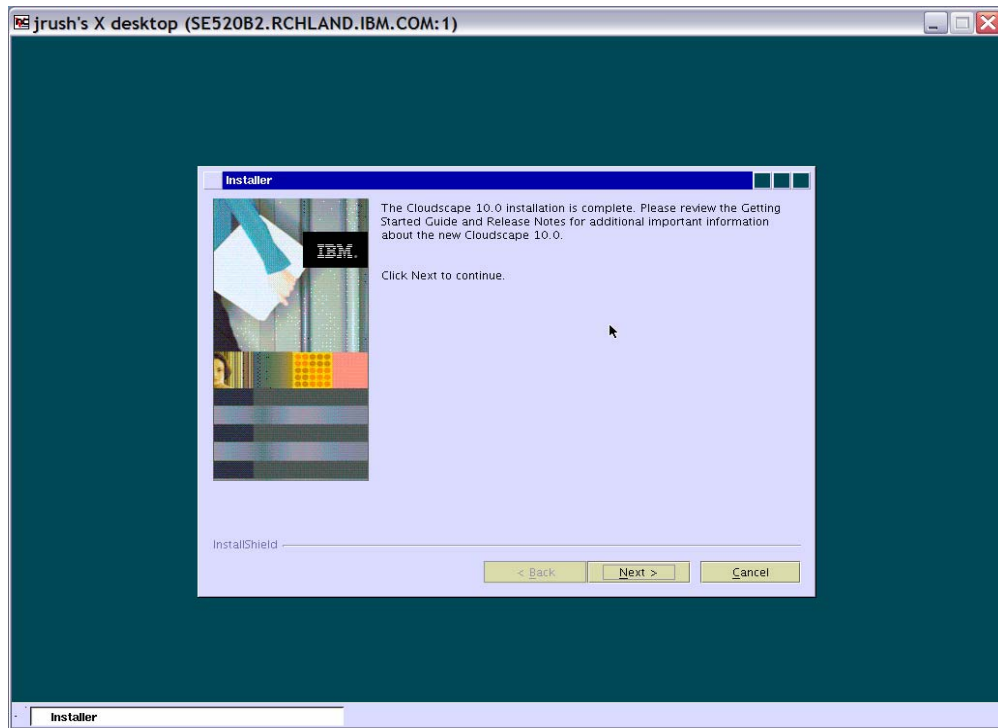


Figure 16: Completion screen

10. When the screen shown in Figure 17 is shown, click **Finish**.

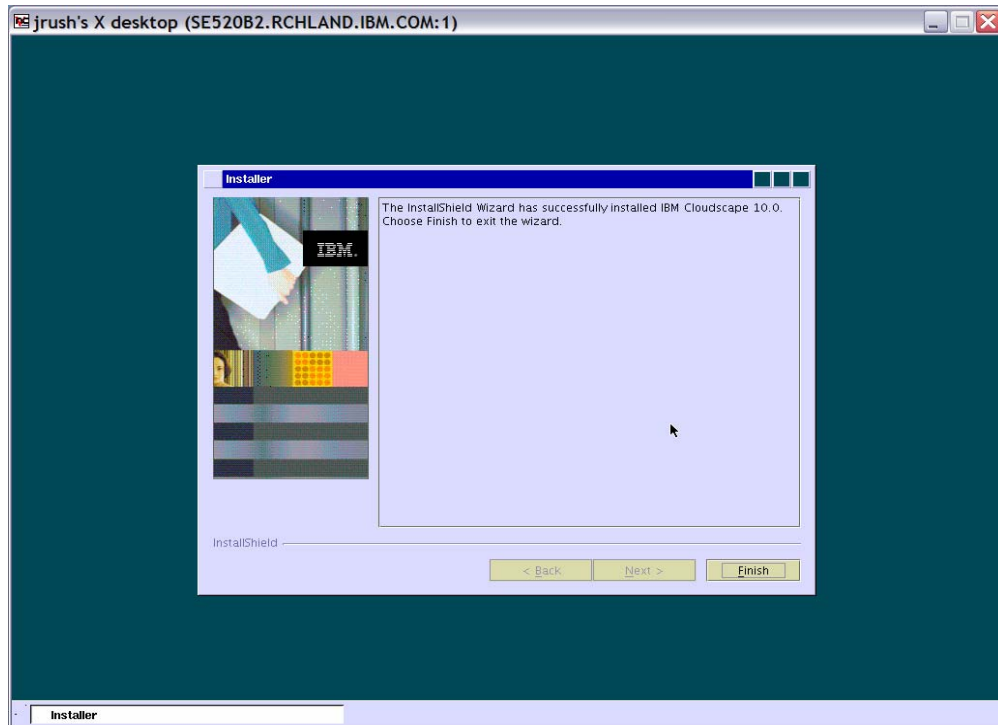


Figure 17: Successful installation

Starting the Cloudscape network server

Now that the Cloudscape server has been installed, start it in networked mode to listen for requests from any IP address. The default TCP/IP port on which Cloudscape listens is 1527.

Entering a QSH environment using the STRQSH command

For Cloudscape V10.0

- Change directory to:
/opt/IBM/Cloudscape_10.0/frameworks/NetworkServer/bin
- Modify the **startNetworkServer.ksh** file as follows for Cloudscape 10.0:

```
# -----
# -- This simple script is an example of how to start Derby
# -- as a server inside the Network Server framework
# --
# -- REQUIREMENTS:
# --     You must have the derby and Network Server jar files in
# --     your CLASSPATH
# --
# -- Check the setNetworkServerCP.ksh file for an example of
# --     what to set.
# --
# -- This file for use on Unix ksh systems
# -----
# -- start Derby Network Server
# -----

CLOUDSCAPE_INSTALL=/opt/IBM/Cloudscape_10.0

[ -z "$CLASSPATH" ] && {
.
"$CLOUDSCAPE_INSTALL"/frameworks/NetworkServer/bin/setNetworkServerCP.ksh
}

# - Original
#java org.apache.derby.drda.NetworkServerControl start
#java org.apache.derby.drda.NetworkServerControl start -h 0.0.0.0 &
# -----
# -- To use a different JVM with a different syntax, simply edit
# -- this file
# -----
```

The modification made in the startup script directs the Cloudscape network server to listen and service requests from an IP address indicated by the **-h 0.0.0.0** option on the **NetworkServerControl** start command: **/startNetworkServer.ksh &**

For Cloudscape V10.1

- Change directory to:
`/opt/IBM/Cloudscape_10.1/frameworks/NetworkServer/bin`
- Export `JAVA_HOME=/QIBM/ProdData/Java400/jdk14`.
- Invoke the `startNetworkServer.ksh` script. The startup script in V10.1 has been updated to take parameters on the invocation of the `startNetworkServer` script. It will accept the host as the first parameter. For example, one of the following:
 - `./startNetworkServer.ksh '0.0.0.0' &`
 - `./startNetworkServer.ksh 'se520b2.rchland.ibm.com' &`

In the first example, `0.0.0.0` indicates that the server will listen on all interfaces and accept connections from remote and local host clients (URLs that use `se520b2.rchland.ibm.com` or `localhost`). The second example indicates that clients will have to explicitly use the hostname (`se520b2.rchland.ibm.com`) in the connection URL. URLs that use `localhost` will be rejected. If a host parameter is not specified, then the network server will only accept connections from URLs that use `localhost` and will also not allow connections from other hosts. The second parameter allows a listening port to be specified other than the default 1527.

Exploring sample Java applications

Now that the Cloudscape server is running, it is possible to execute some Java samples to create a database, create a table and insert data, and query the contents. These samples can be cut, pasted, and modified for a particular environment. They can be compiled and run in another iSeries QShell window with the IBM DB2 Universal Database™ (UDB) universal JDBC driver jar files in the CLASSPATH of the iSeries JVM. Otherwise, run these from a client that has the DB2® UDB jar files. These jar files are: `db2jcc.jar`, `db2jcc_license_cu.jar`, or `db2jcc_license_cisuz.jar`. In these examples, modify sections that are surrounded by `<>`. For example, replace `<hostname>` with the TCP/IP hostname of the iSeries system.

The code below will connect to the Cloudscape network server listening on port 1527. The <user>, <password>, and <hostname> in the code below are the iSeries user, password, and the TCP/IP hostname of the iSeries system respectively. The database that is being created, <somedatabasename>, can be any name desired.

```
import java.sql.CallableStatement;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.Statement;
import java.util.Properties;

public class i5CloudscapeJDBCCreate {

    public static void main(String[] args) {
        Statement stmt = null;
        Connection connection = null;
        CallableStatement cs = null;
        Statement cp = null;

        String driver = "com.ibm.db2.jcc.DB2Driver";
        System.out.println("START");
        try {
            Class.forName(driver);
            Properties properties = new java.util.Properties();

            properties.setProperty("user", "<userid>");
            properties.setProperty("password", "<password>");
            connection = DriverManager.getConnection(
                "jdbc:derby:net://<hostname>:1527/<somedatabasename>;create=true",
                properties);
            connection.close();

            System.out.println("CREATED");
        } catch (Throwable e) {
            System.out.println("Caught exception:");
            e.printStackTrace(System.out);
        }
    }
}
```

Creating a table and inserting rows into the database

The code below will create a table named EMPLOYEE and will insert two rows into that table.

```
import java.sql.CallableStatement;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.Properties;

public class i5CloudscapeJDBCInsert {

    public static void main(String[] args) {
        Statement stmt = null;
        Connection connection = null;
        CallableStatement cs = null;
        Statement cp = null;

        String driver = "com.ibm.db2.jcc.DB2Driver";

        try {
            Class.forName(driver);
            Properties properties = new java.util.Properties();
            properties.setProperty("user", "<userid>");
            properties.setProperty("password", "<password>");

            connection = DriverManager.getConnection(
                "jdbc:derby:net://<hostname>:1527/<somedatabasename>", properties);
            try {
                Statement s = connection.createStatement();
                /*
                 * We create a table, add a few rows, and update one.
                 */
                s.execute("create table employee(fname char(20), lname char(20), empid int)");
                System.out.println("Created table employee");
                s.execute("insert into employee values ('Jon', 'Rush', 1958001)");
                System.out.println("Inserted Jon Rush 1958001");
                s.execute("insert into employee values ('Joe', 'Black', 1962002)");
                System.out.println("Inserted Joe Black 1962002");

                s.close();
                connection.close();

            } catch (SQLException e) {
                System.out.println("Error code:" + e.getErrorCode());
            }

        } catch (Throwable e) {
            System.out.println("Caught exception:");
            e.printStackTrace(System.out);
        }
    }
}
```


Take note that if a schema is not specified on the *create table employee* statement, a schema will be created (or used) that matches the user value in the properties object. In this case the schema will be JRUSH. In the example, to explicitly specify a schema, the create statement would be:

```
create table <someschema>.employee(fname char(20), lname char(20), empid int)
```

where <someschema> is an existing schema in the jerdb2 database.

Query the database

The code below will query the employee table in the database, selecting all records.

```
import java.sql.CallableStatement;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.Properties;

public class i5CloudscapeJDBC {

    public static void main(String[] args) {
        Statement stmt = null;
        Connection connection = null;
        CallableStatement cs = null;
        Statement cp = null;

        String driver = "com.ibm.db2.jcc.DB2Driver";

        try {
            Class.forName(driver);
            Properties properties = new java.util.Properties();
            properties.setProperty("user", "<userid>");
            properties.setProperty("password", "<password>");

            connection = DriverManager.getConnection(
                "jdbc:derby:net://<hostname>:1527/<somedatabasename>", properties);
            String sql = "select * from employee";
            try {
                stmt = connection.createStatement();
                System.out.println("Execute select");
                ResultSet rs = stmt.executeQuery(sql);
                while (rs.next()) {
                    System.out.println(rs.getObject(1));
                    System.out.println(rs.getObject(2));
                }
                stmt.close();
                connection.close();
            } catch (SQLException e) {
                System.out.println("Error code:" + e.getErrorCode());
            }

        } catch (Throwable e) {
            System.out.println("Caught exception:");
            e.printStackTrace(System.out);
        }

    }
}
```

Using the Cloudscape interactive tools

Several interactive tools are available for application development when using the Cloudscape database. The following is a brief description of the Derby interactive JDBC (ij) scripting tool and the Rational Application Development (RAD) tool.

Using the Derby interactive JDBC scripting tool

The ij tool is Derby's interactive JDBC scripting tool. It is a simple utility for running scripts. The ij tool is actually a Java application that is started from a command window such as an MS-DOS® command window or the UNIX® shell. It provides several commands for easily accessing a variety of JDBC features through the developer's use of scripts.

- From the i5/OS QShell environment change the directory to: **IBM/Cloudscape_10.1/frameworks/NetworkServer/bin**. This is where the network server started and ran the Java samples to create the jcrdb2 database. Looking at the IFS directory structure, notice a directory under *bin* named **jcrdb2**. This is the physical repository for the jcrdb2 database.
- Enter the following on the command line: **./setNetworkServerCP.ksh** and press **Enter**. Notice at the beginning of this command that there are two periods (separated from each other by a space). Do not forget the first period as it sets up the CLASSPATH that enables the ij tool to run in the next step.
- Enter this command next: **java org.apache.derby.tools.ij**

Invoking ij options

Additional ij invocation options are available, as shown below:

```
java [<options>] org.apache.derby.tools.ij [-p <propertyFile>] [<inputFile>]
```

The command line items for the above invocation are:

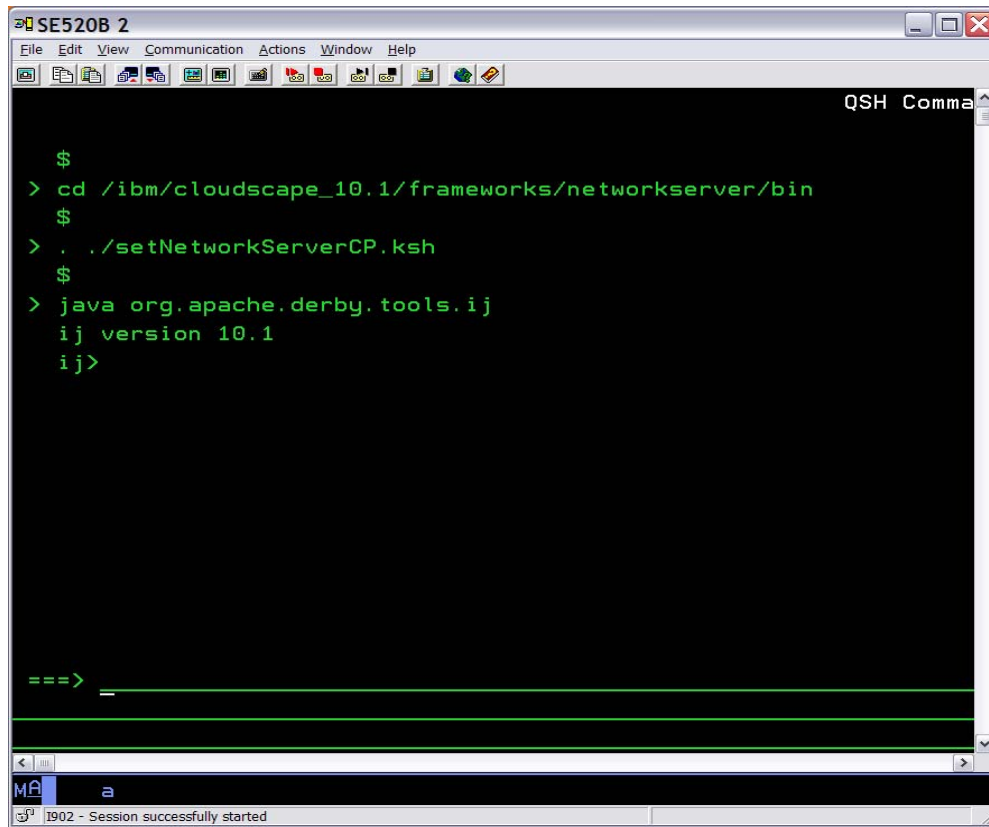
- **java**: The name of the JVM program
- **options**: The options that will be used by the JVM. Invoke the -D option to set ij properties or system properties, such as Derby properties
- **propertyFile**: A file used to set ij properties (instead of the -D option). The property file needs to be in the format created by the Java.tools.Properties.save methods, which is the same format as the derby.properties file.
- **inputFile**: A file from which commands are read. The ij tool exits either at the end of the file or as the result of an exit command. Using an input file causes ij to print the commands as it runs them. If the standard input is

rerouted, ij does not print the commands. If an input file is not supplied, ij reads from the standard input.

Creating a schema, table, and data with ij

The following figure shows that the ij environment has been started and is awaiting commands (Figure 18).

NOTE: all commands entered into the ij environment must end with a ‘;’.



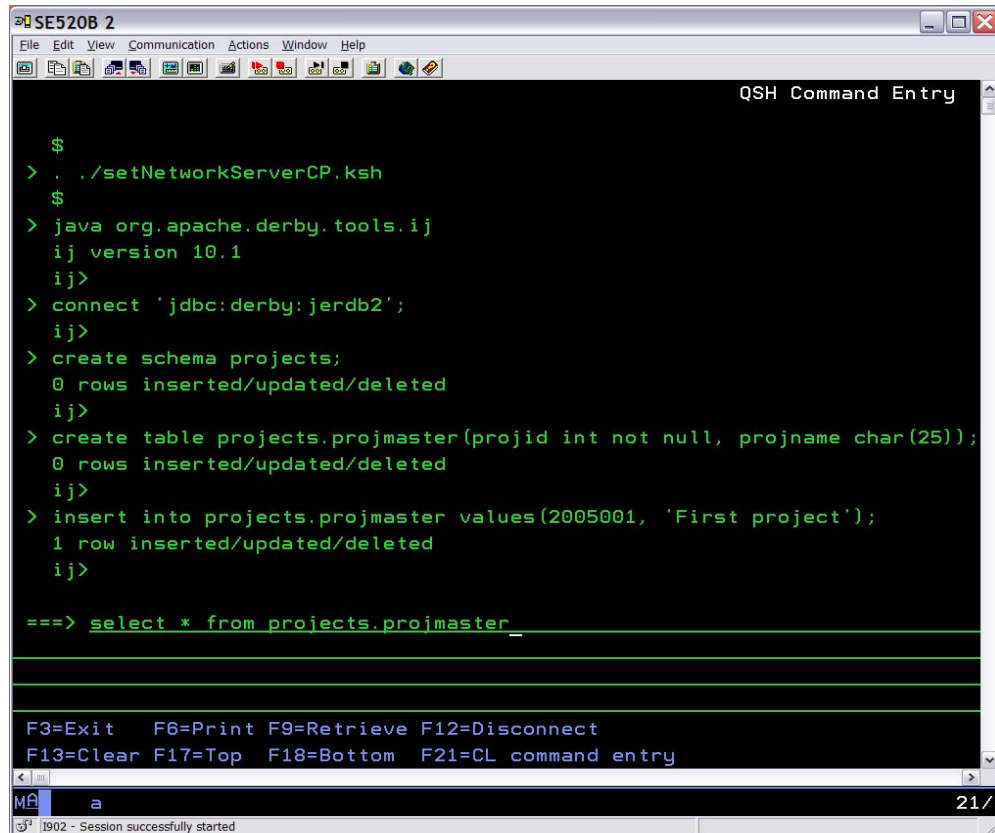
The screenshot shows a terminal window titled 'SE520B 2' with a menu bar (File, Edit, View, Communication, Actions, Window, Help) and a toolbar. The terminal content is as follows:

```
QSH Comma
$
> cd /ibm/cloudscape_10.1/frameworks/networkserver/bin
$
> . ./setNetworkServerCP.ksh
$
> java org.apache.derby.tools.ij
ij version 10.1
ij>
```

At the bottom of the terminal, there is a prompt '===>' followed by a horizontal line. Below the terminal window, a status bar shows 'MA a' and '1902 - Session successfully started'.

Figure 18: ij environment started

Now connect to the jerdb2 database and create a schema named **PROJECTS** (Figure 19).



```
SE520B 2
File Edit View Communication Actions Window Help
QSH Command Entry

$
> . ./setNetworkServerCP.ksh
$
> java org.apache.derby.tools.ij
ij version 10.1
ij>
> connect 'jdbc:derby:jerdb2';
ij>
> create schema projects;
0 rows inserted/updated/deleted
ij>
> create table projects.projmaster(projid int not null, projname char(25));
0 rows inserted/updated/deleted
ij>
> insert into projects.projmaster values(2005001, 'First project');
1 row inserted/updated/deleted
ij>

==> select * from projects.projmaster_

F3=Exit F6=Print F9=Retrieve F12=Disconnect
F13=Clear F17=Top F18=Bottom F21=CL command entry
MA a 21/0
I902 - Session successfully started
```

Figure 19: Create a schema named *PROJECTS*

Accessing and deploying a Cloudscape database using RAD

The RAD tool provides a Rational Software Development Platform for rapid construction of applications that access Cloudscape databases.

Using an existing schema

Start the RAD tool, open the data perspective, and create a new database connection as follows:

1. Use the IBM universal JDBC driver, class `com.ibm.db2.jcc.DB2Driver`, to connect to the network server on the iSeries system. Include the DB2 UDB license jar files as well as the `db2jcc` jar file that contains the `DB2Driver` class in the class location field.
2. Make sure the Connection URL is structured as in Figure 20. “se520b2” is the TCP/IP hostname of the iSeries system and `jerdb2` is the name of the database that was previously created with the sample code (Figure 21).

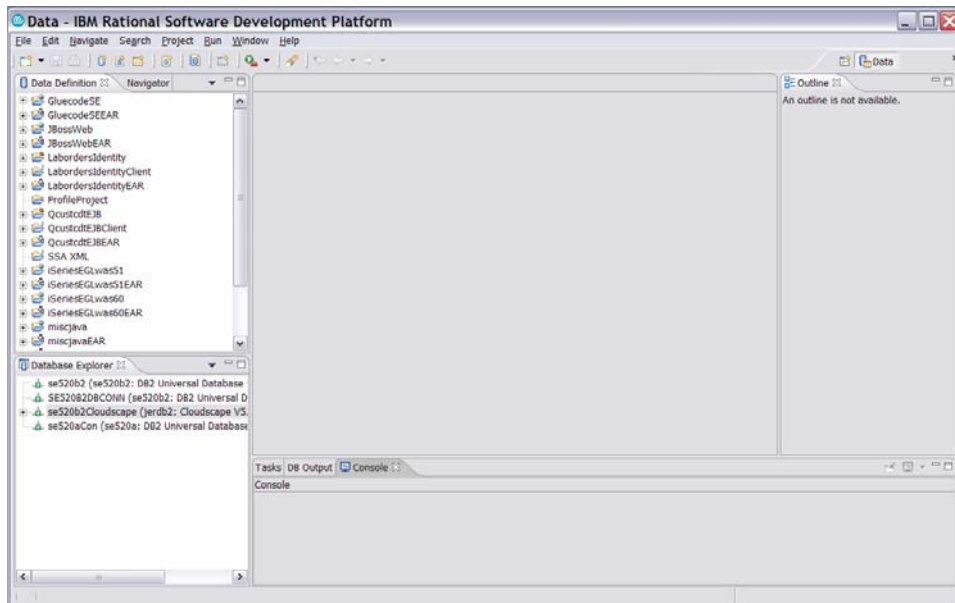


Figure 20: Connection URL

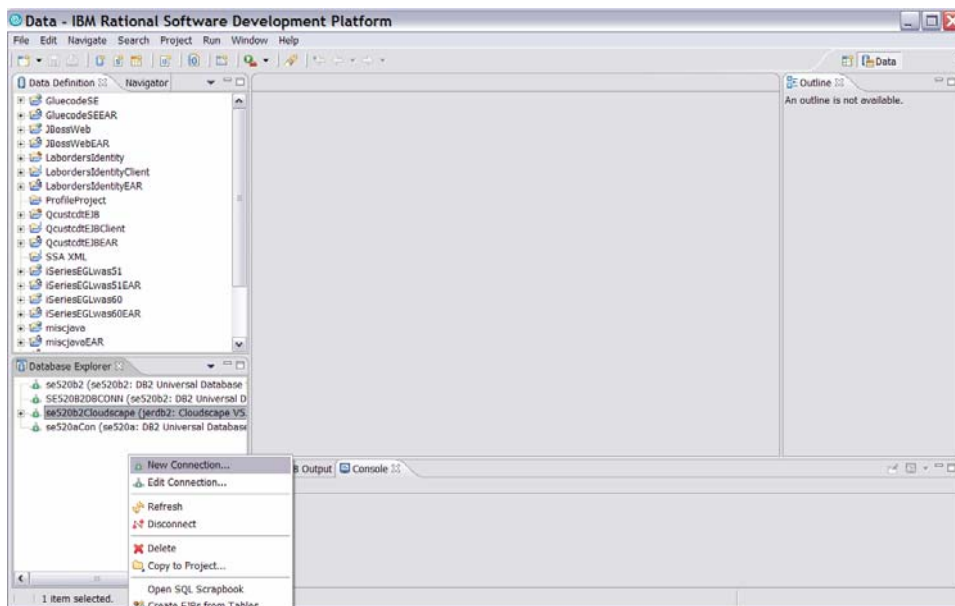


Figure 21: TCP/IP hostname

3. From the **Specify connection parameters** screen (Figure 22) click **Finish**.

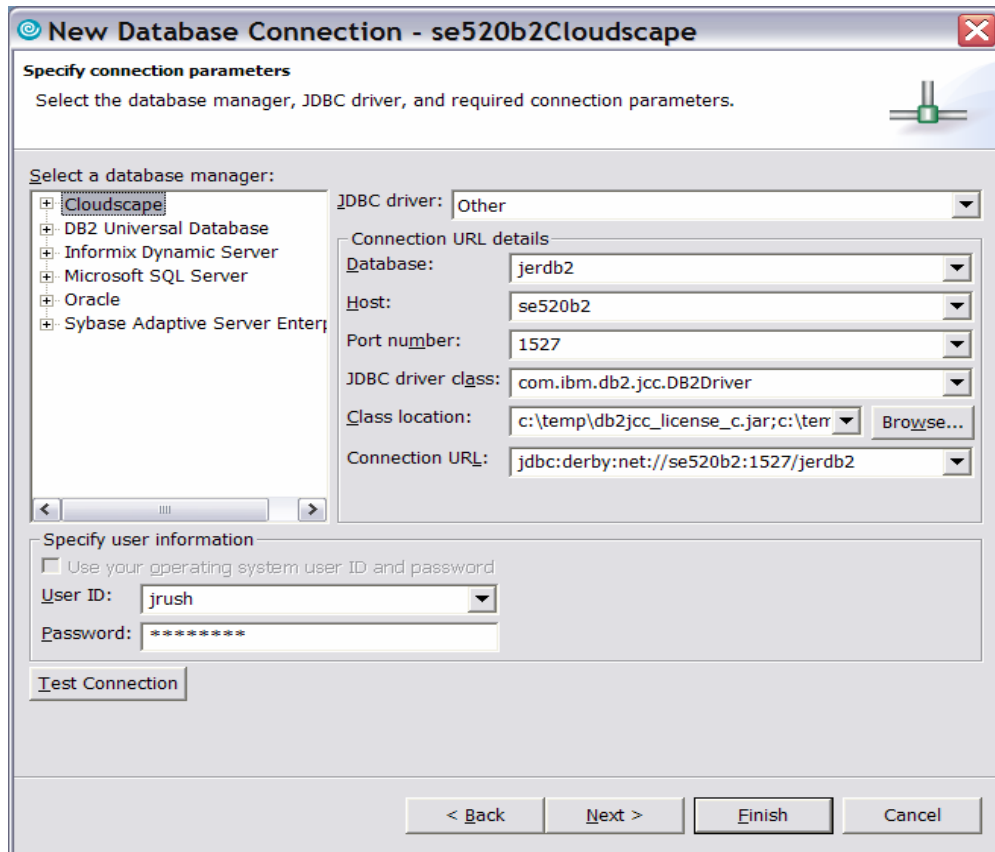


Figure 22: Specify connection parameters screen

4. An error might pop up. Click **OK**.

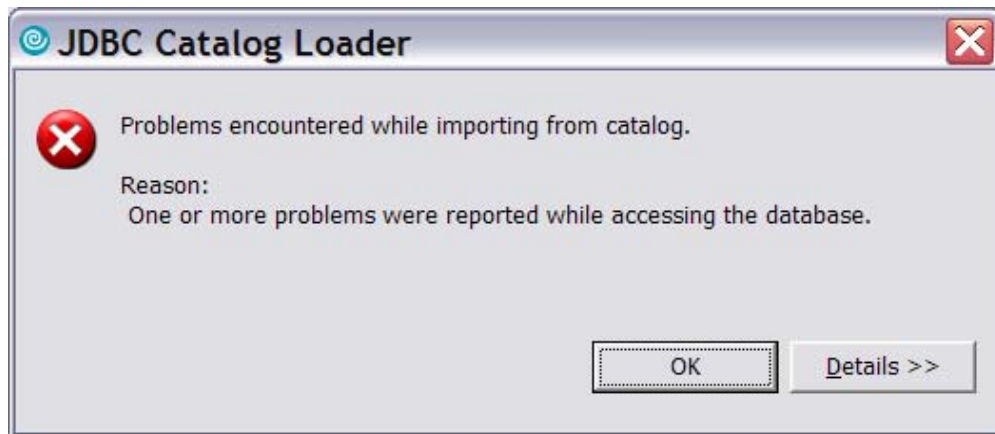


Figure 23: Error notice

5. Drill down into the Cloudscape database to find the employee table created in the previous section (Figure 24).

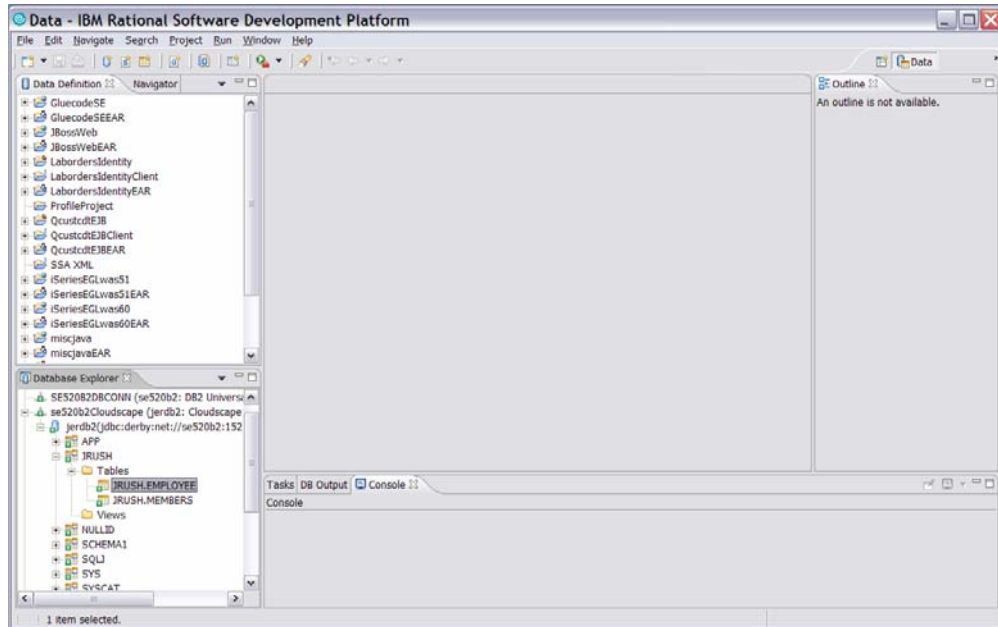


Figure 24: Employee table

6. Right-click the table and select **Sample Contents** as circled in red below (Figure 25):

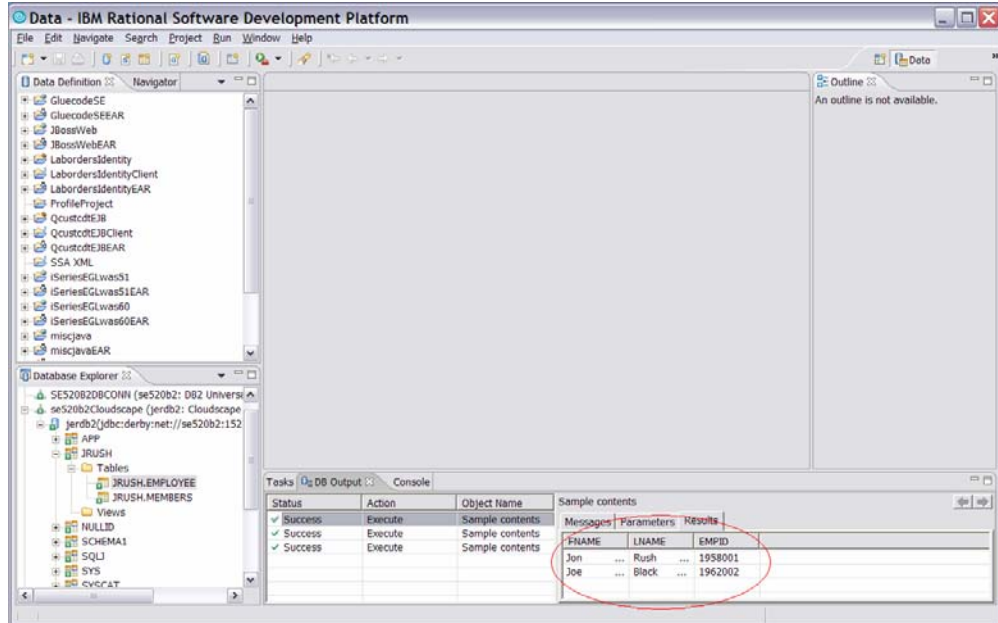


Figure 25: Select Sample Contents

Using the interactive SQL Scrapbook

The data perspective of the workbench also has an interactive SQL capability where SQL commands are executed against the iSeries Cloudscape database connection.

1. In the Database Explorer pane select the active connection to the iSeries Cloudscape database. This is circled in red in Figure 26 below.
2. Right-click it and select **Open SQL Scrapbook** (Figure 27).

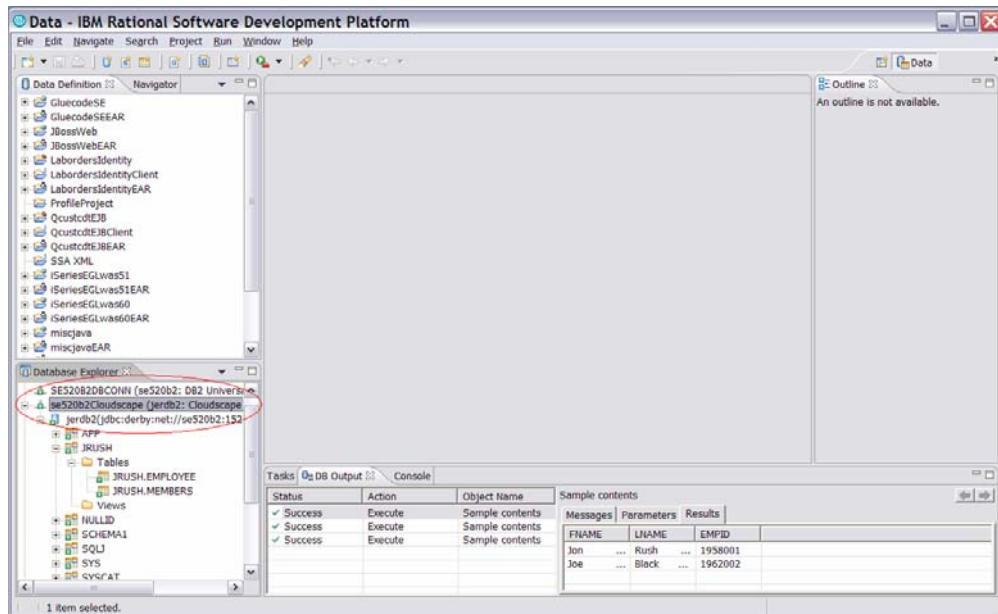


Figure 26: Select active connection

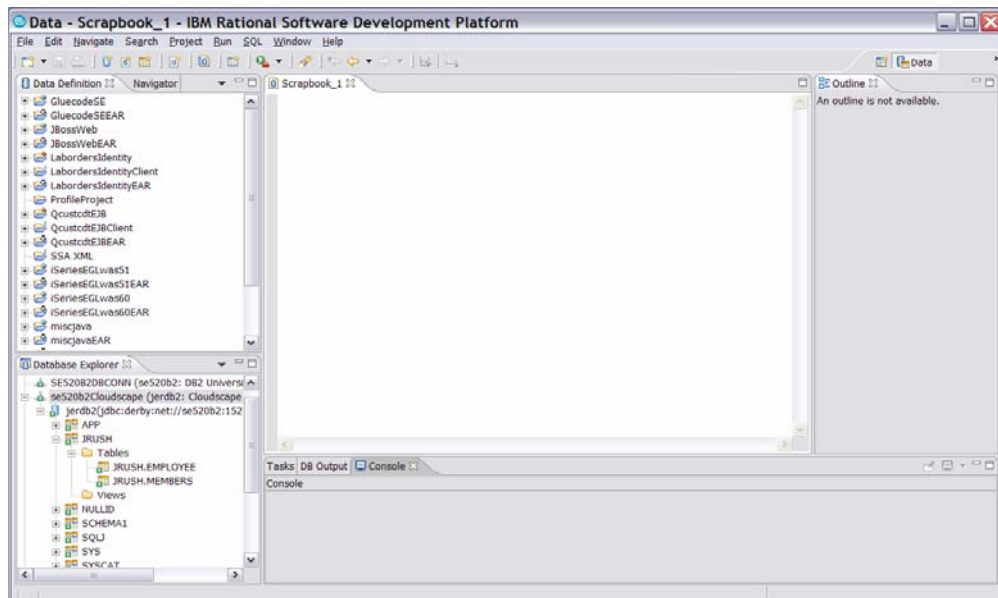


Figure 27: Open SQL Scrapbook

3. When the SQL Scrapbook pane opens, input and execute SQL commands. For example, to query the contents of the employee table, enter or select an SQL statement as shown in Figure 28. Notice that the statement has qualified the employee table to be in the jrush schema. Then, from the SQL menu, select **Execute**. The bottom right pane should show the results of the query.

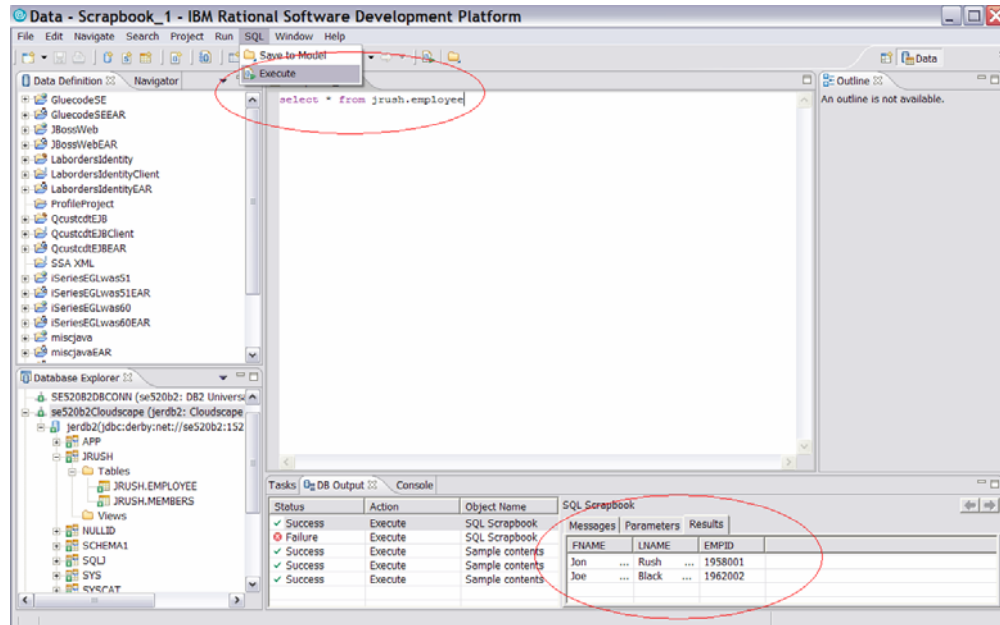


Figure 28: Execute SQL commands

4. Insert a row into the employee table. Enter the insert statement as shown below in Figure 29.

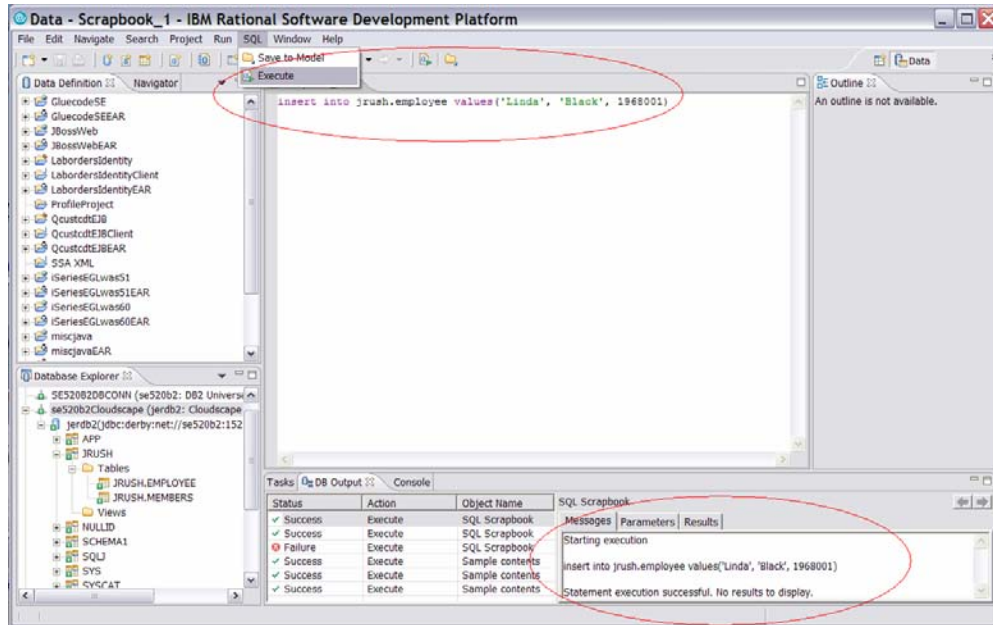


Figure 29: Insert statement

Using RAD to create a new schema and table

1. In the data perspective, right-click a **project** and select **New > Database Definition** (Figure 30).

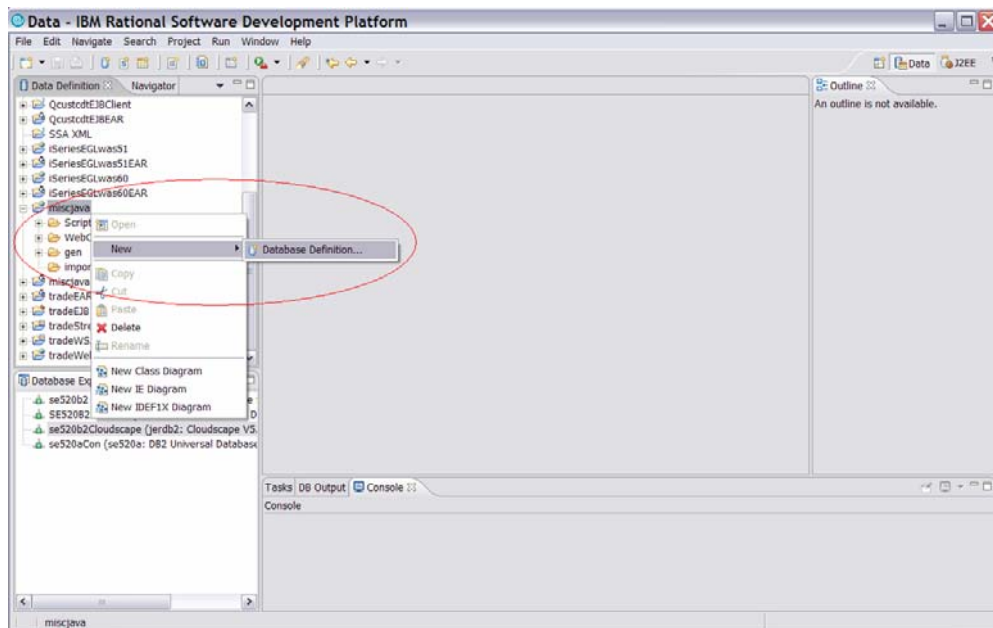


Figure 30: Select Database Definition

2. The database name field should be the database created in the sample Java code. In the *Database vendor type* field, select **Cloudscape V5.1** and press **Finish** (Figure 31).

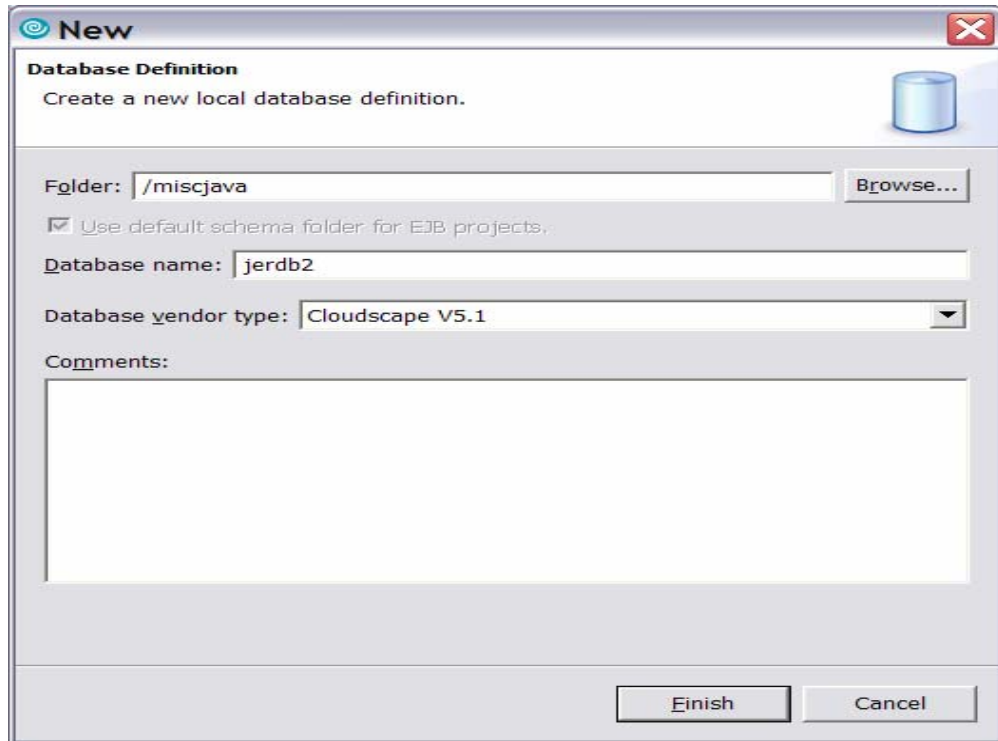


Figure 31: Select Database vendor type

3. The *Data Definition* pane on the right will be updated to include the *JERDB2 (Cloudscape V5.1)* structure under the project selected above (circled in red in Figure 32).

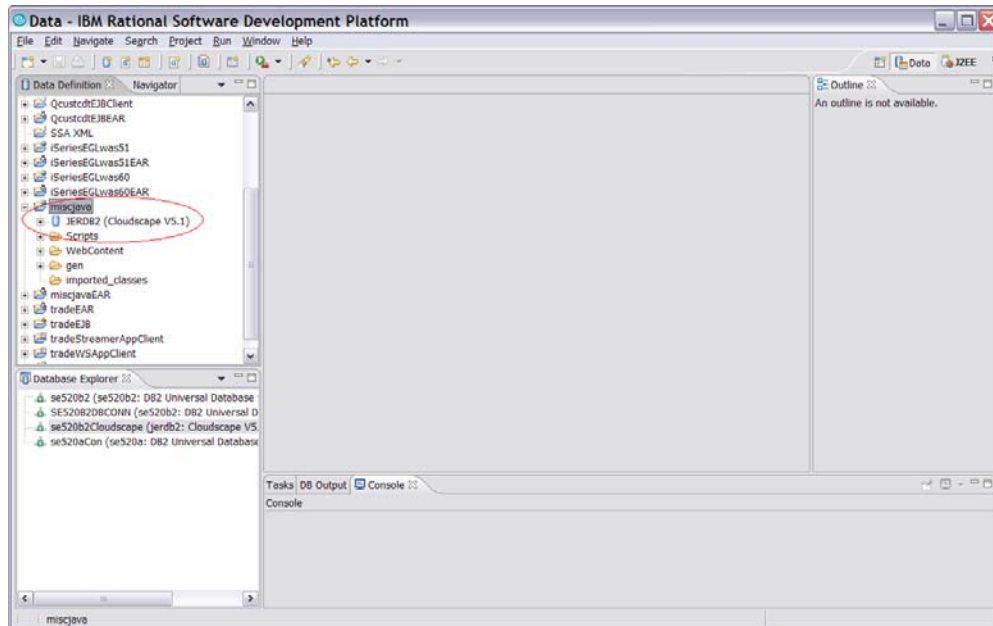


Figure 32: Updated Data Definition pane

4. Create a schema by right-clicking **JERDB2 (Cloudscape V5.1)** and selecting **Run > Schema Definition** (circled in red below in Figure 33).

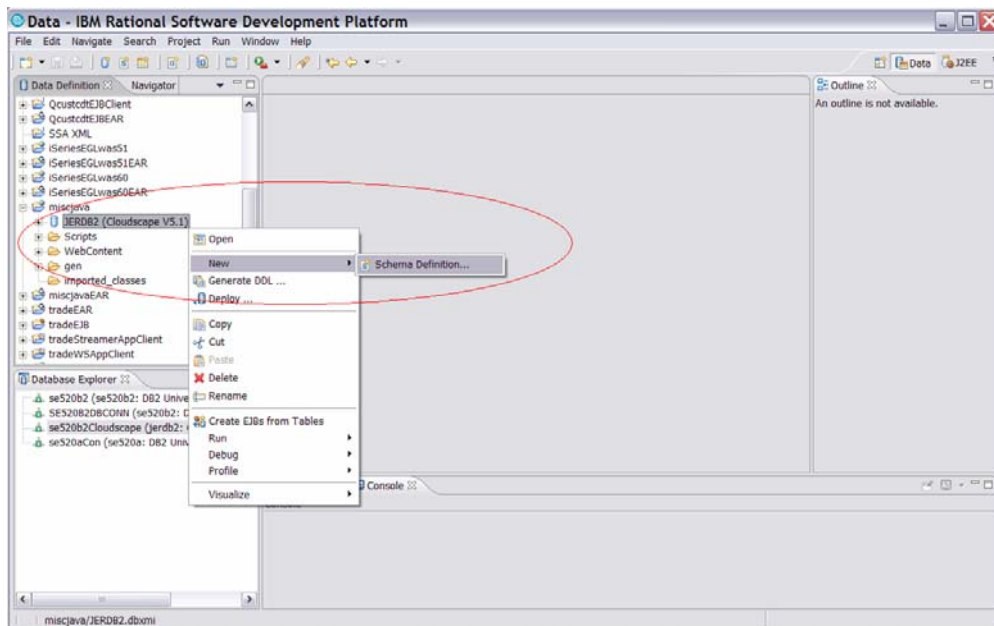


Figure 33: Create a schema

5. Click **Finish** (Figure 34).

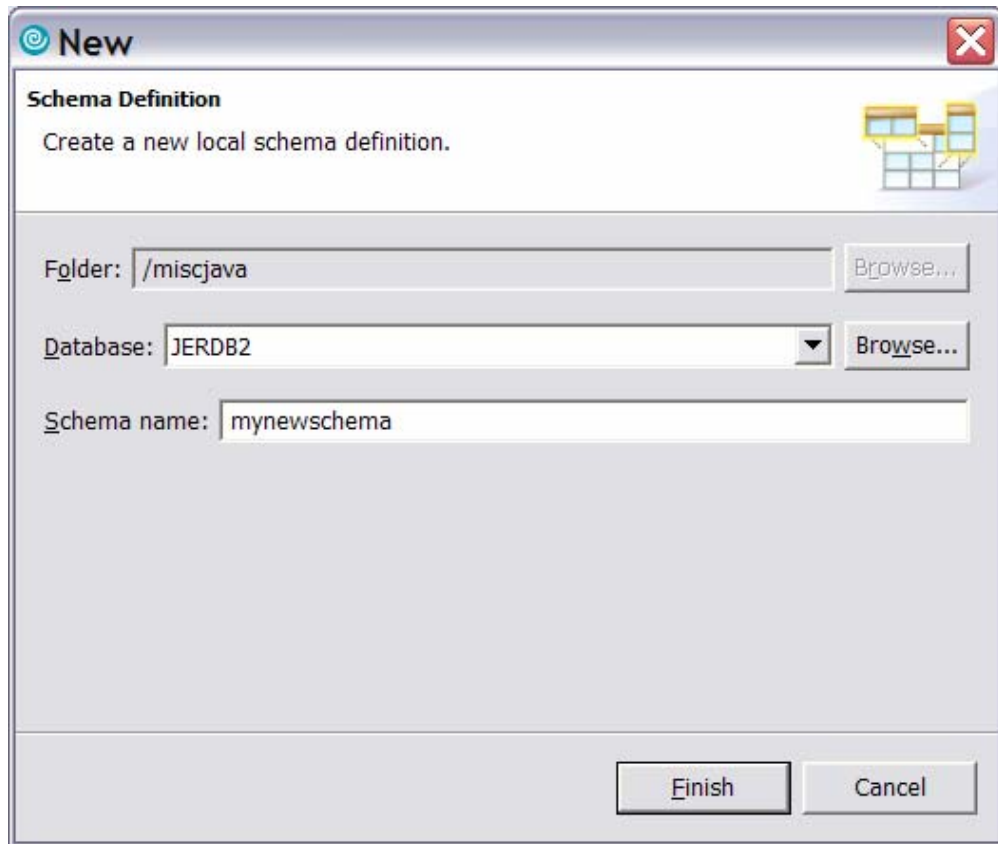


Figure 34: Finish schema definition

6. The *Data Definition* pane should now look something like Figure 35.

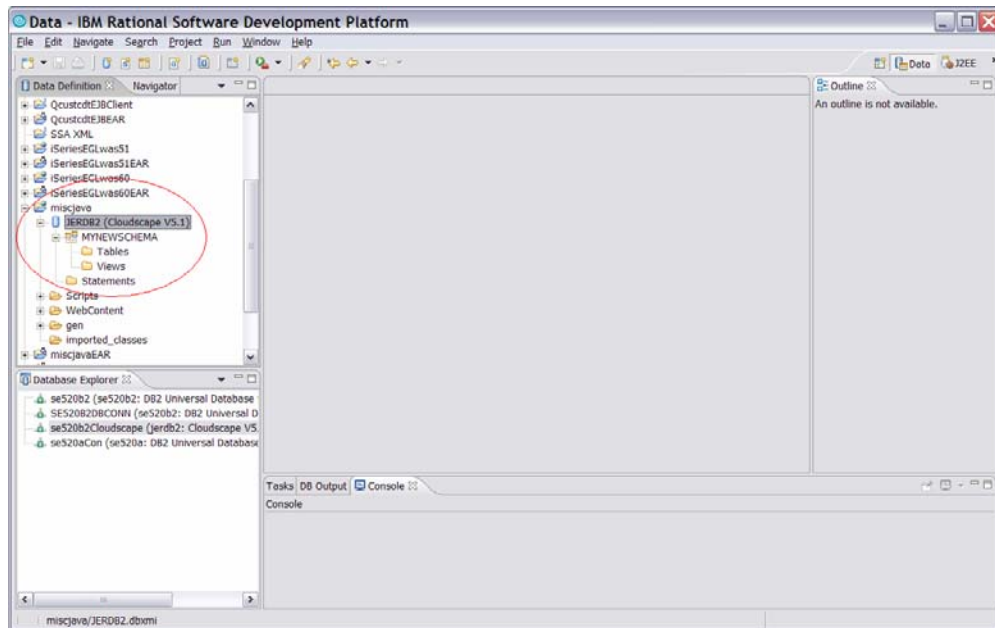


Figure 35: Updated Data Definition

7. Create a new table definition by clicking **Tables > New > Table Definition** (Figure 36).

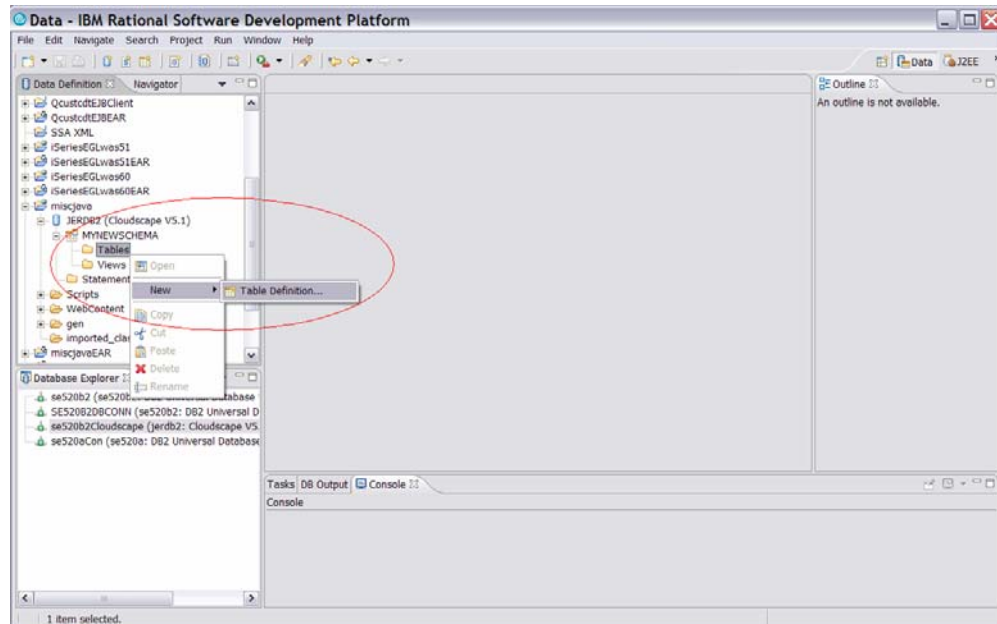
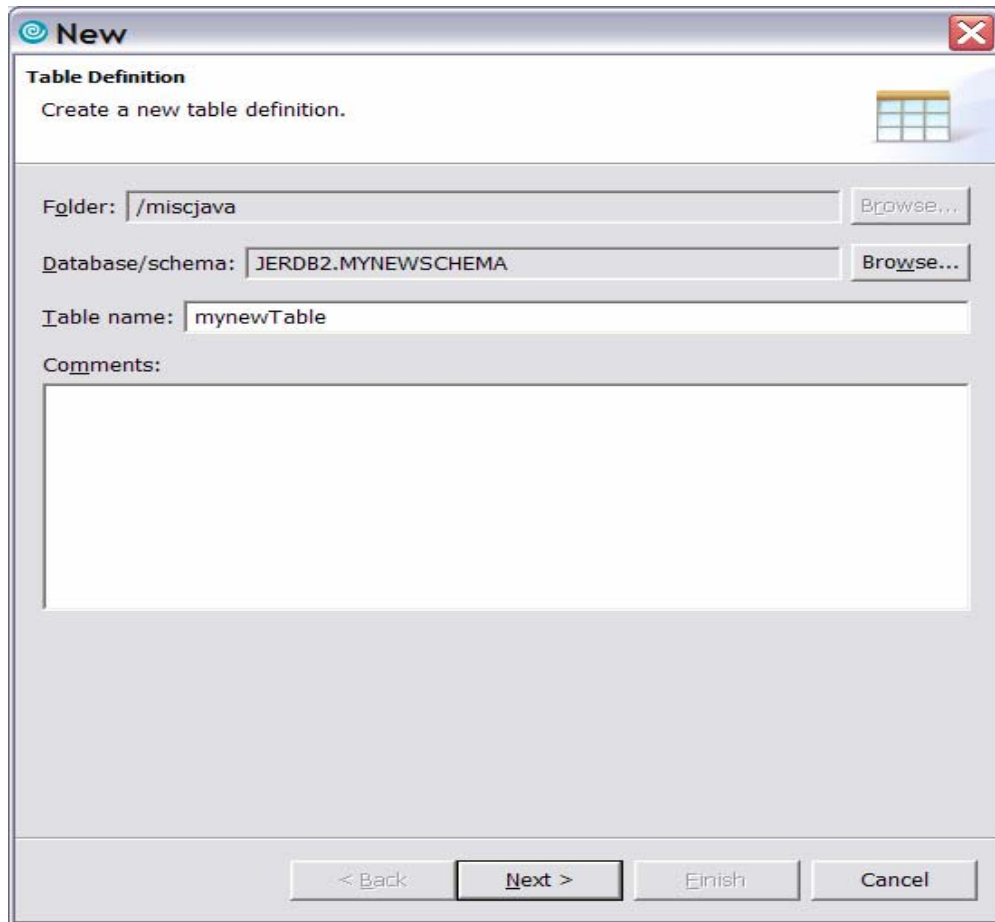


Figure 36: Create table definition

8. Click **Next** (Figure 37).



The screenshot shows a dialog box titled "New" with a close button in the top right corner. The main heading is "Table Definition" with a sub-instruction "Create a new table definition." and a small table icon. Below this, there are three input fields: "Folder:" containing "/miscjava" with a "Browse..." button; "Database/schema:" containing "JERDB2.MYNEWSHEMA" with a "Browse..." button; and "Table name:" containing "mynewTable". A "Comments:" label is followed by a large empty text area. At the bottom, there are four buttons: "< Back", "Next >" (which is highlighted), "Finish", and "Cancel".

Figure 37: Enter table definition data

9. Click **Add Another** to add a column to the table definition (Figure 38).

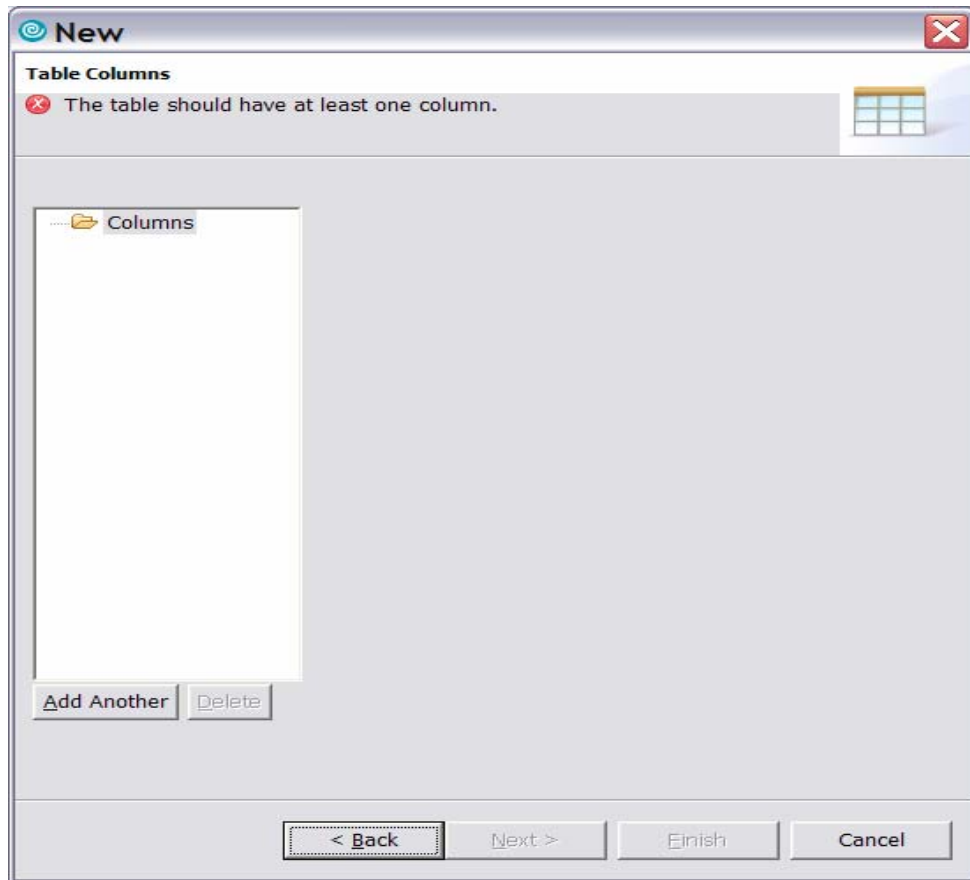


Figure 38: Add column to table definition

10. Click **Add Another** (Figure 39).

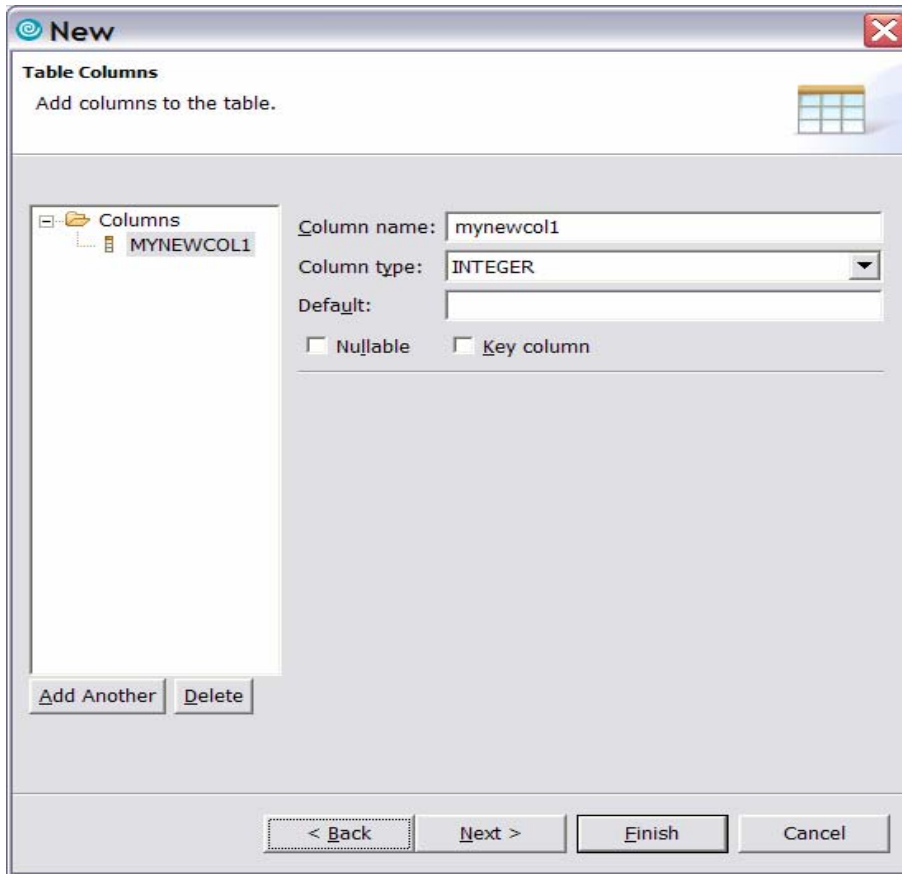


Figure 39: Add another column

11. Click **Finish** (Figure 40).

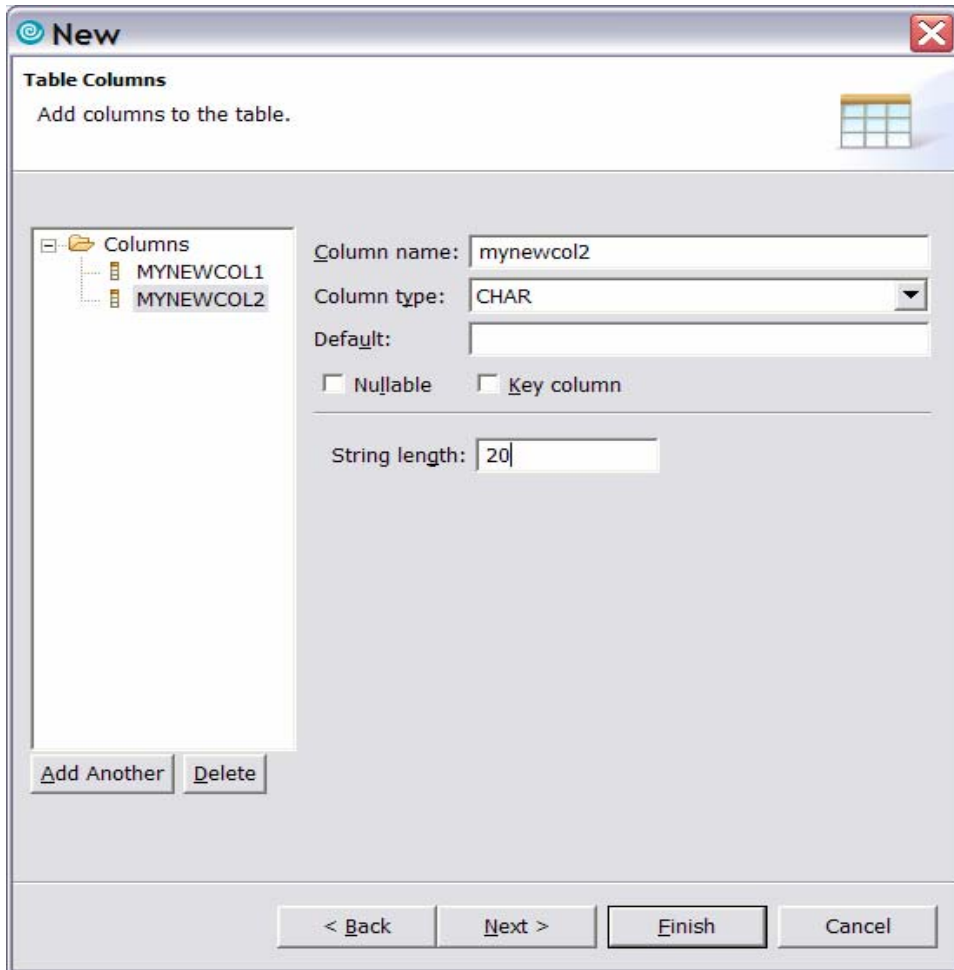


Figure 40: Finish table columns

12. The *Data Definition* pane will look like this (circled in red in Figure 41):

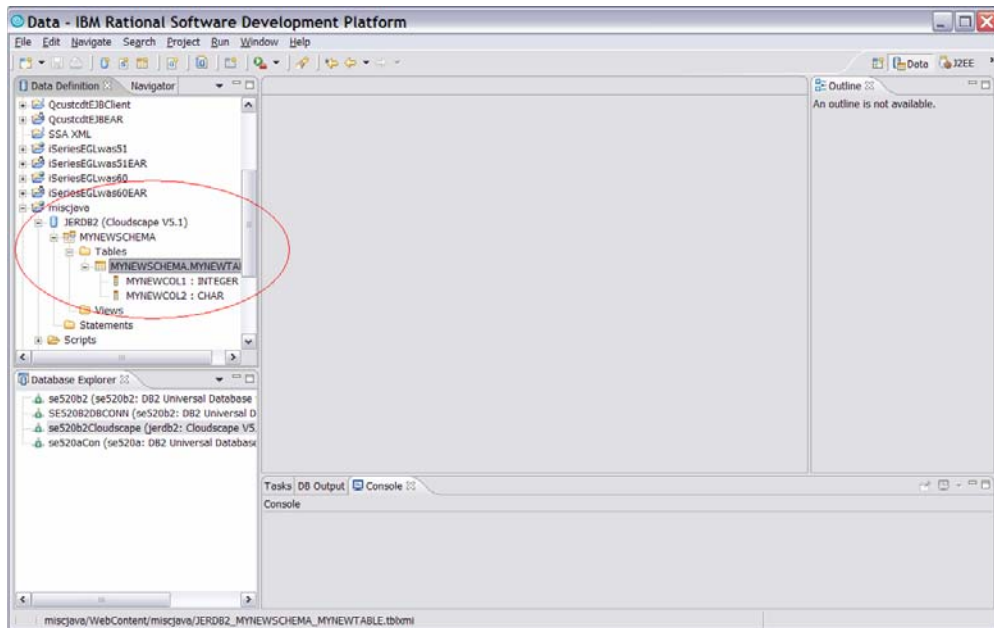


Figure 41: Updated Data Definition

13. Now deploy to the Cloudscape server on the iSeries system. Right-click the server name, and click **Deploy** (Figure 42).

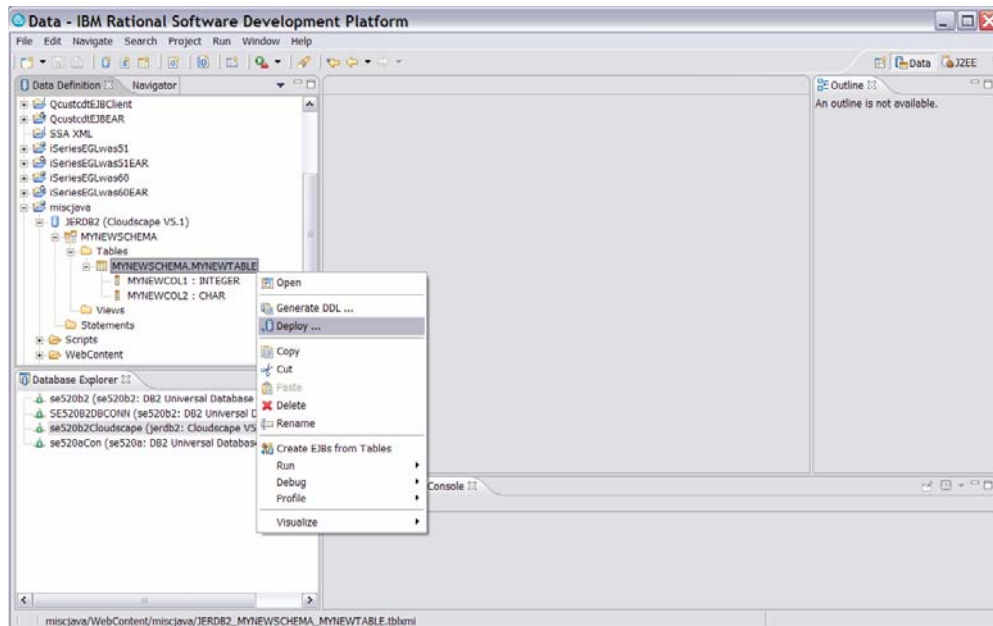


Figure 42: Deploy to Cloudscape sever

14. You will see the following screen in you will select the table(s) to deploy. Then click **Next** (Figure 43).

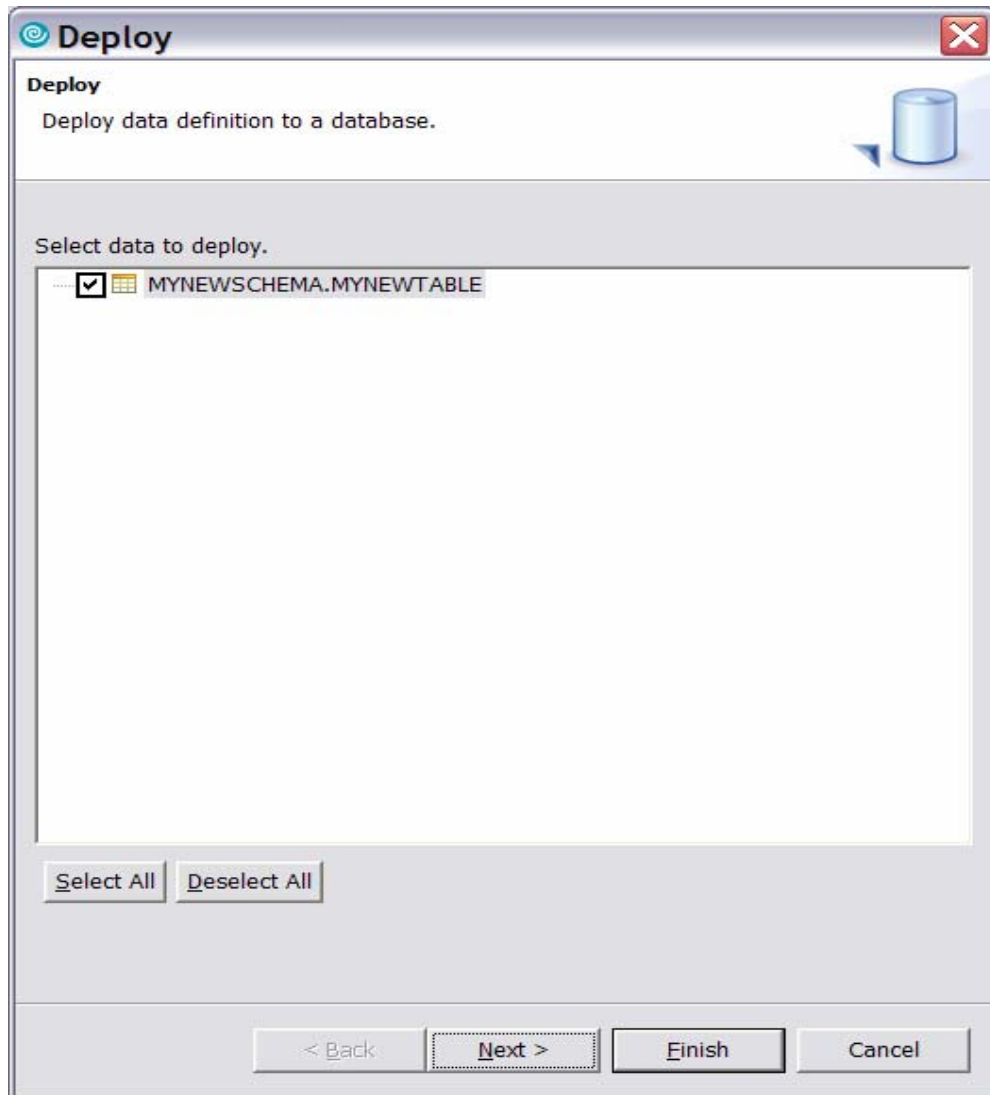


Figure 43: Select data to deploy

15. From the *Data Export Options* screen, click **Next** (Figure 44).



Figure 44: Choose data export options

16. On the *Establish a JDBC connection to a database* screen click the box to choose **Use existing connection**. Click **Finish** (Figure 45).

Deploy

Database Connection
Establish a JDBC connection to a database.

Connection name: Con5

Database: jerdb2

User ID: jrush

Password:

Database vendor type: DB2 Universal Database V8.2

JDBC driver: IBM DB2 UNIVERSAL DRIVER

Host: se520b2

(Optional) Port number: 50000

Server name:

Database Location: Browse...

JDBC driver class: com.ibm.db2.jcc.DB2Driver

Class location: C:\Program Files\IBM\SQLLIB\java\db2jcc4\classes Browse...

Connection URL: jdbc:derby:net://se520b2:1527/jerdb2

Filters...

Use existing connection:
Existing connection: se520b2Cloudscape

< Back Next > Finish Cancel

Figure 45: Establish JDBC connection

Resources

These Web sites provide useful reference materials to supplement the information contained within this document.

- IBM Cloudscape home page
ibm.com/software/data/cloudscape
- Linux on POWER Cloudscape information
ibm.com/servers/enable/linux/power/pdfs/cloudscape_lop.pdf
- Apache Software Foundation Derby project home page
<http://db.apache.org/derby>
- IBM Cloudscape requirements
ibm.com/software/data/cloudscape/requirements.html
Provides a list of specific releases for various platforms and Java environments for which Cloudscape has been tested
- IBM Cloudscape Version 10.0 (includes download information)
ibm.com/developerworks/subscriptions/descfiles/icdscv10.htm
- IBM Cloudscape Version 10.1 (includes download information)
ibm.com/developerworks/db2/library/techarticle/dm-0408cline
- Cloudscape FAQs
ibm.com/developerworks/db2/library/techarticle/dm-0408bradbury/index.html
- IBM Rational home page
ibm.com/software/rational
- VNC viewer download
www.realvnc.com/download.html
- IBM Redbooks™
ibm.com/redbooks
- iSeries Information Center
<http://publib.boulder.ibm.com/infocenter/series/v5r3/ic2924/index.htm?info/icmain.htm>

Trademarks and special notices

© IBM Corporation 1994-2005. All rights reserved.

References in this document to IBM products or services do not imply that IBM intends to make them available in every country.

The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both: IBM, the IBM logo, eServer, iSeries, i5/OS, Cloudscape, DB2, DB2 UDB, Redbooks, and ibm.com.

Rational is a trademark of International Business Machines Corporation and Rational Software Corporation in the United States, other countries, or both.

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

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

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

Information is provided "AS IS" without warranty of any kind.