

dbConnector Getting Started – Version 2.1.2.19211

This Getting Started Guide provides you with two test scenarios that should acquaint you with the features of dbConnector. By following each of the steps in this Guide you should be able to create a searchable dtSearch index complete with custom fields.

What is dbConnector?

dbConnector allows you to define relationships between your unstructured data (pdfs, Word documents, etc.) and structured data (data in databases) and generate a dtSearch index complete with database information ('custom fields'). The resulting index allows you to perform more sophisticated searches, provides opportunities for advanced filtering and sorting capabilities, and unifies your business information in one searchable repository.

Now, you may ask the question: "How can it help me?" Consider some examples within the context of the test scenarios we provide with this Getting Started document. In the test scenarios we have provided a database with a table for Suppliers. Additionally we have documents with product specifications for the Supplier's products. We may, for example, wish to search the product specification documents and then get in touch with a contact person at the Supplier if we like what we see or have additional questions. In our scenario, the product specifications don't have contact names for each of the Suppliers, the database does, however. So, we'll use dbConnector to define the relationship between the documents and the database and identify that we want the ContactName in the Supplier table added to the index.

Once we've created the index we can sort the search results by the contact person, filter results by saying I only want to list products for a given contact, etc. Or, another query might be "I only want to deal with product suppliers located in London".

This is but one hypothetical example of how dbConnector can be deployed. Feel free to contact us at support@cybergroup.com if you'd like to discuss your database/document project.

Installing dbConnector and Related Test Files

To install dbConnector unzip the contents of the file 'dbconnector_setup.zip'. Included in the zip is the dbConnector installer, setup.exe. Double click this file to install dbConnector.

Create a folder on your C: drive and name it 'dbconnector_test'. Then unzip the accompanying zip file, test_scenarios.zip into the 'dbconnector_test' folder.

When you unzip the files you will have three folders created beneath the c:\dbconnector_test directory:

- 1) c:\dbconnector_test\database
- 2) c:\dbconnector_test\testdocs
- 3) c:\dbconnector_test\index

Scenario One

The first scenario provided assumes that you have a collection of folders with documents related to records in a database. The database tracks our Suppliers (Suppliers Table). Each record has, among other columns, the CompanyName and ContactName, and SupplierID. This scenario assumes that we have a collection of documents related to each Supplier in a given folder. Perhaps our intention is to create an index of the Supplier's documents that includes the Supplier's ContactName from the database.

Our objective is to create an index of the documents in the folders, and, very importantly, include custom fields from the database in the index. In this scenario we assume that the folder names contain the KEY. The KEY is a unique identifier of the records in the database. In this test case, the KEY is the Supplier ID. We'll 'carry along' the custom fields of CompanyName and ContactName when we create the index.

If you have installed the test files as described above, you should have three folders:

C:\dbconnector_test\testdocs

This directory contains ten folders. Each folder is named 1 through 10 to match a unique Supplier ID in the database. Each of the 10 folders contains several documents (pdfs) that are associated with the Supplier in the database. In our test case we're saying that these are product documents related to a given Supplier.

C:\dbconnector_test\database

In this folder we have a database, testdb.mdb which has two tables, Suppliers and Files (the latter table is used only in scenario two).

C:\dbconnector_test\index

This is a folder which you should have created during the installation above. It will be the destination directory for the index created by dbConnector.

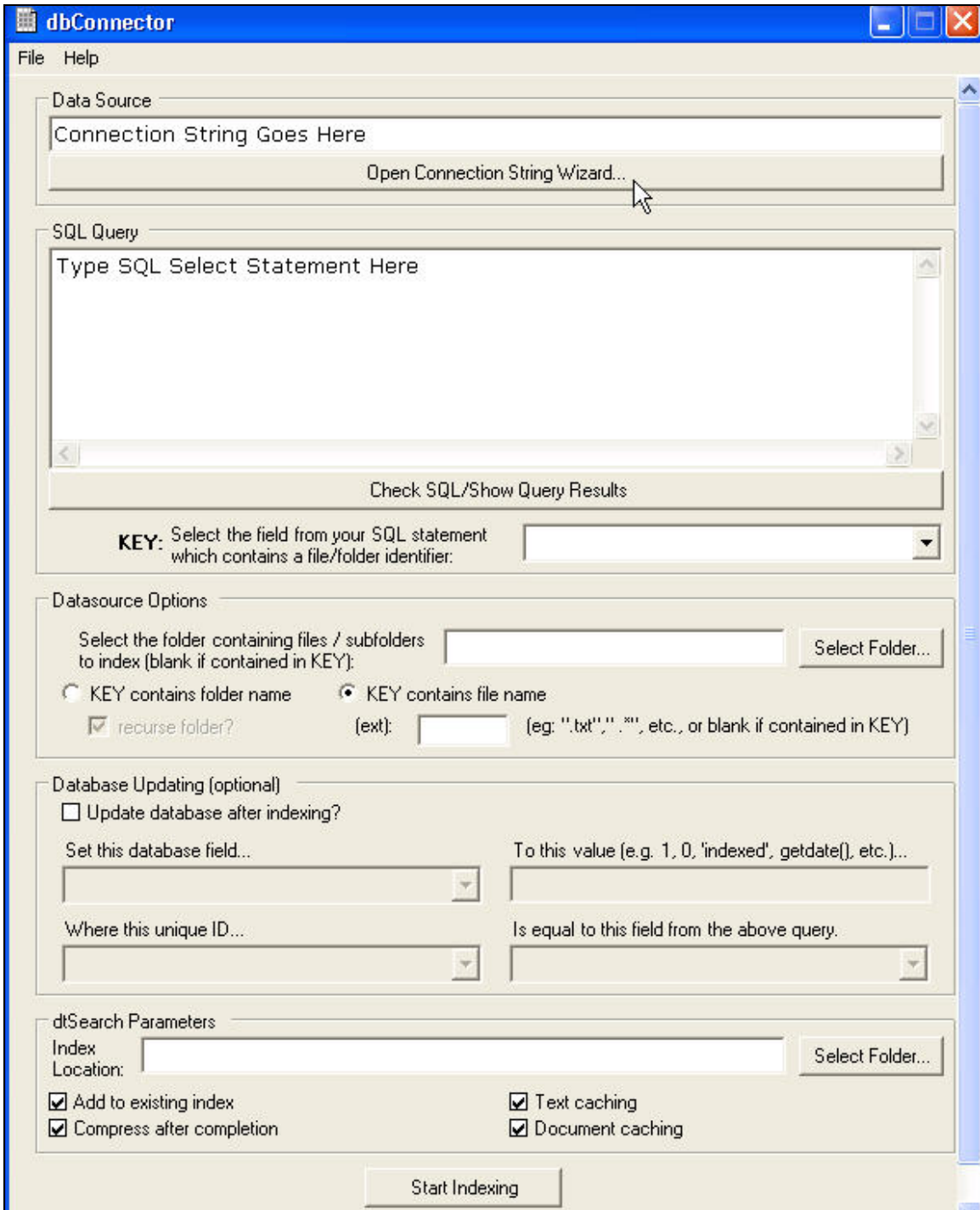


Figure 1

Once you start dbConnector (from the Windows Start Menu / Programs / Cybergroup / dbConnector) the first screen you'll see is shown in Figure 1. Press "Open Connection String Wizard".

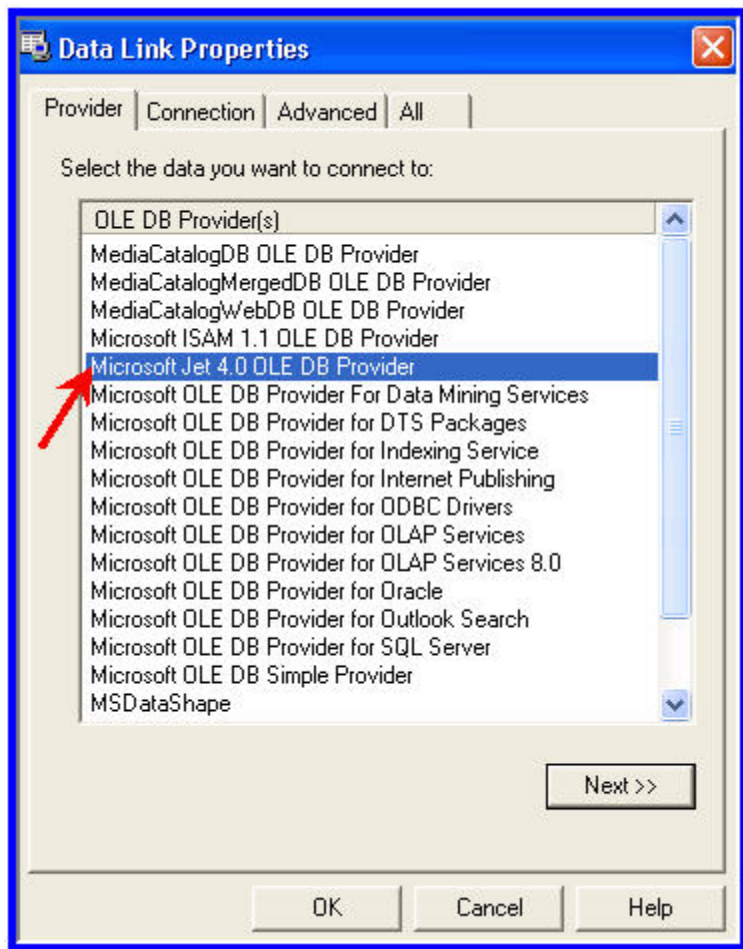


Figure 2

You will now be shown a list of possible OLE DB Providers that you may use to connect dbConnector to your database (Figure 2). For the purposes of the test scenario select 'Microsoft Jet 4.0 OLE DB Provider' and click the 'Next>>' button.

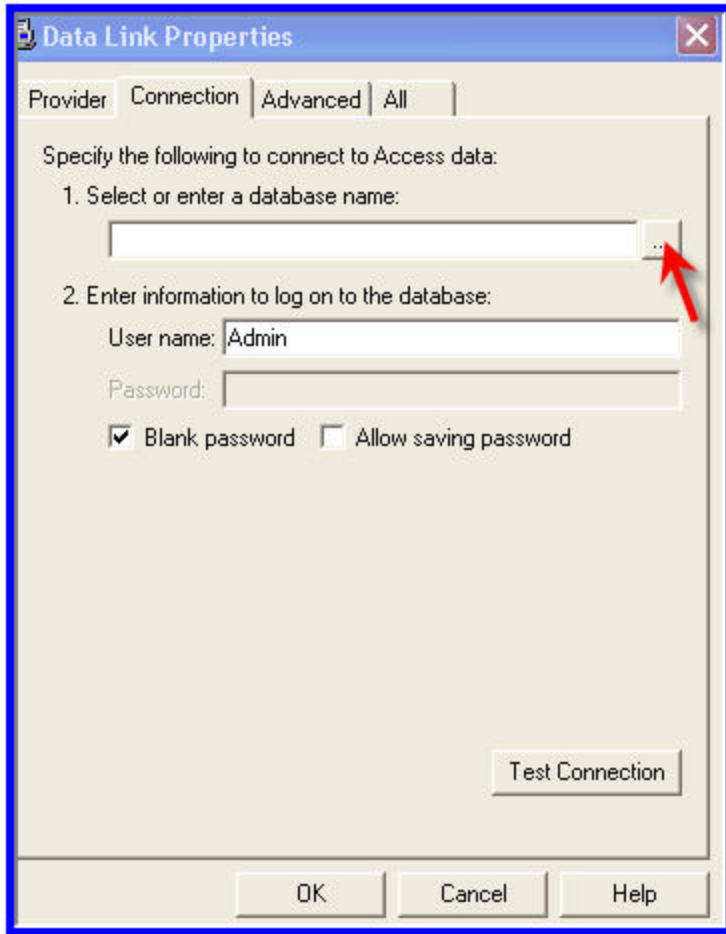


Figure 3

The window shown in Figure 3 will now appear. Press the “...” button to browse for our test database. The database location should be C:\dbConnector_test\database. The filename is testdb.mdb. Find the file and click “Open”.

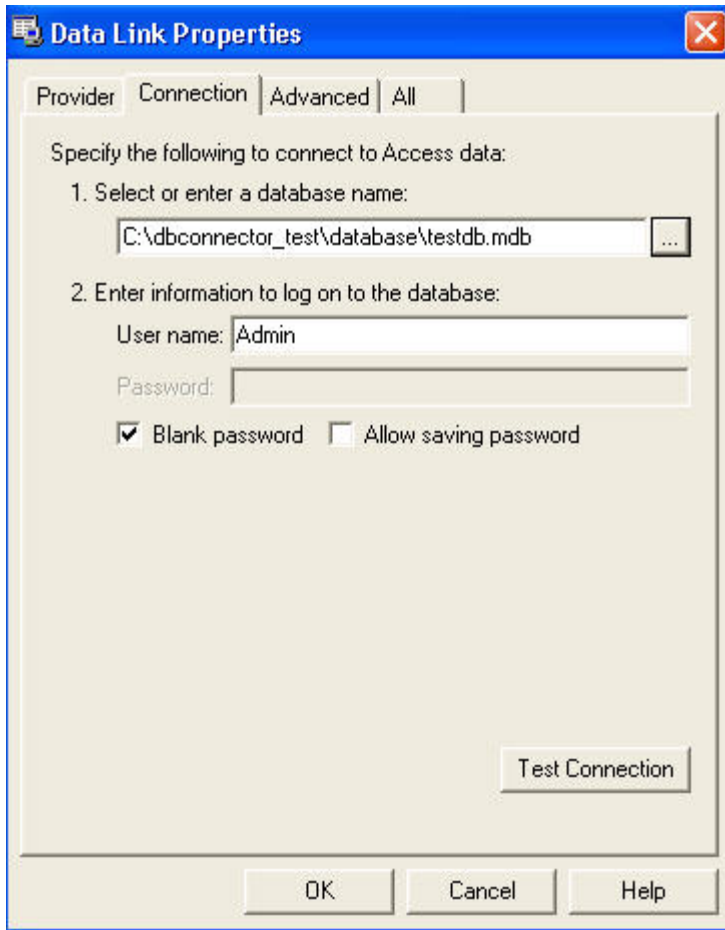


Figure 4

You will then see the screen show in Figure 4. Click "OK".

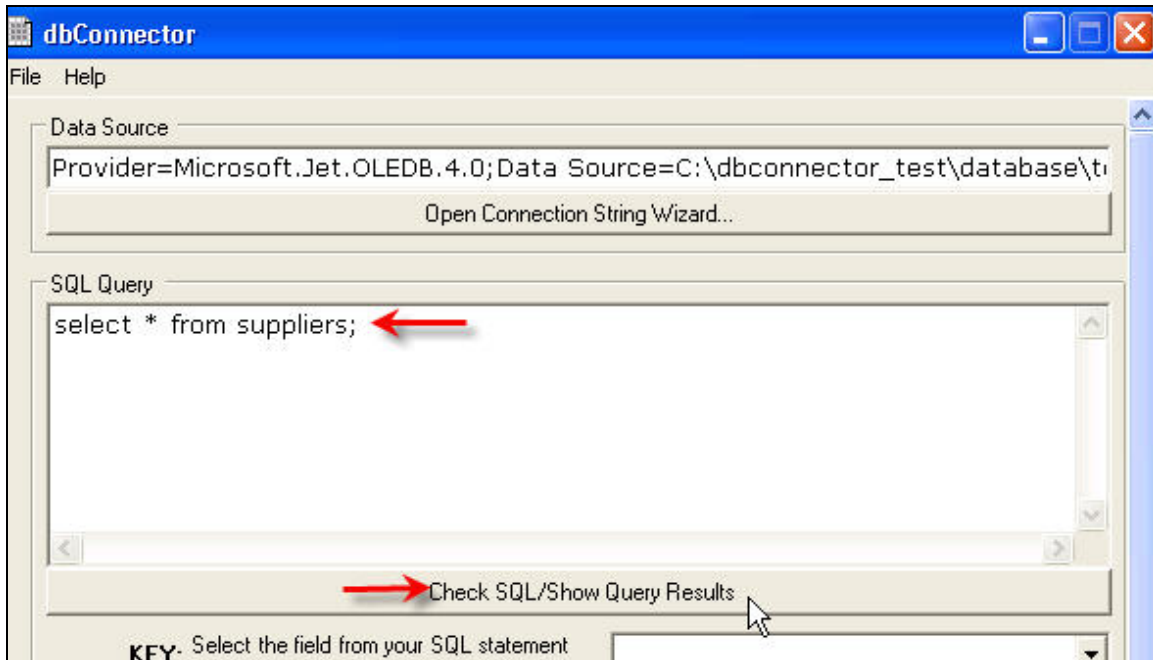


Figure 5

Now, for this scenario, enter the SQL query:

select * from Suppliers;

in the SQL Query window. Then click "Check SQL/Show Query Results". See figure 5.

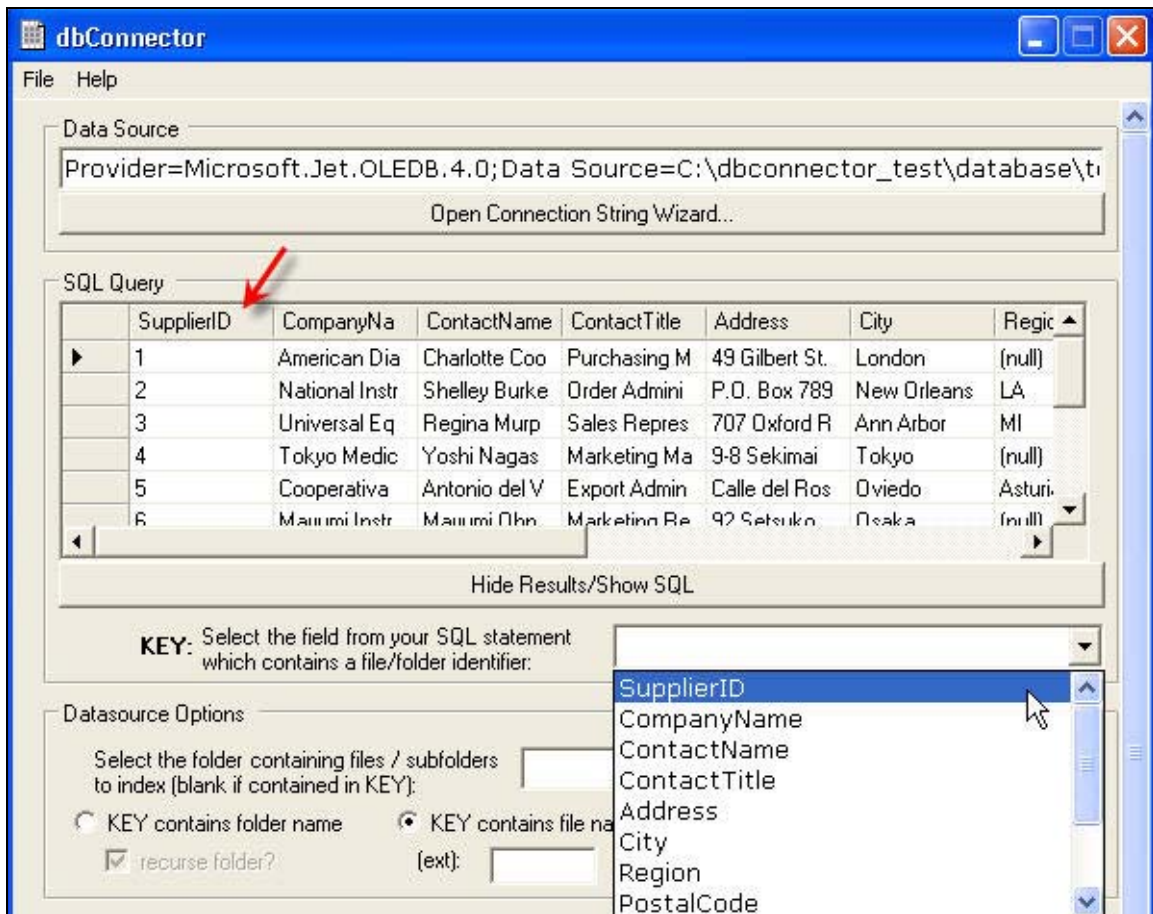


Figure 6

Now you can see the database information. Where it says **KEY:** you must choose the column that contains the unique identification KEY, which, in this case, is SupplierID. (See figure 6)

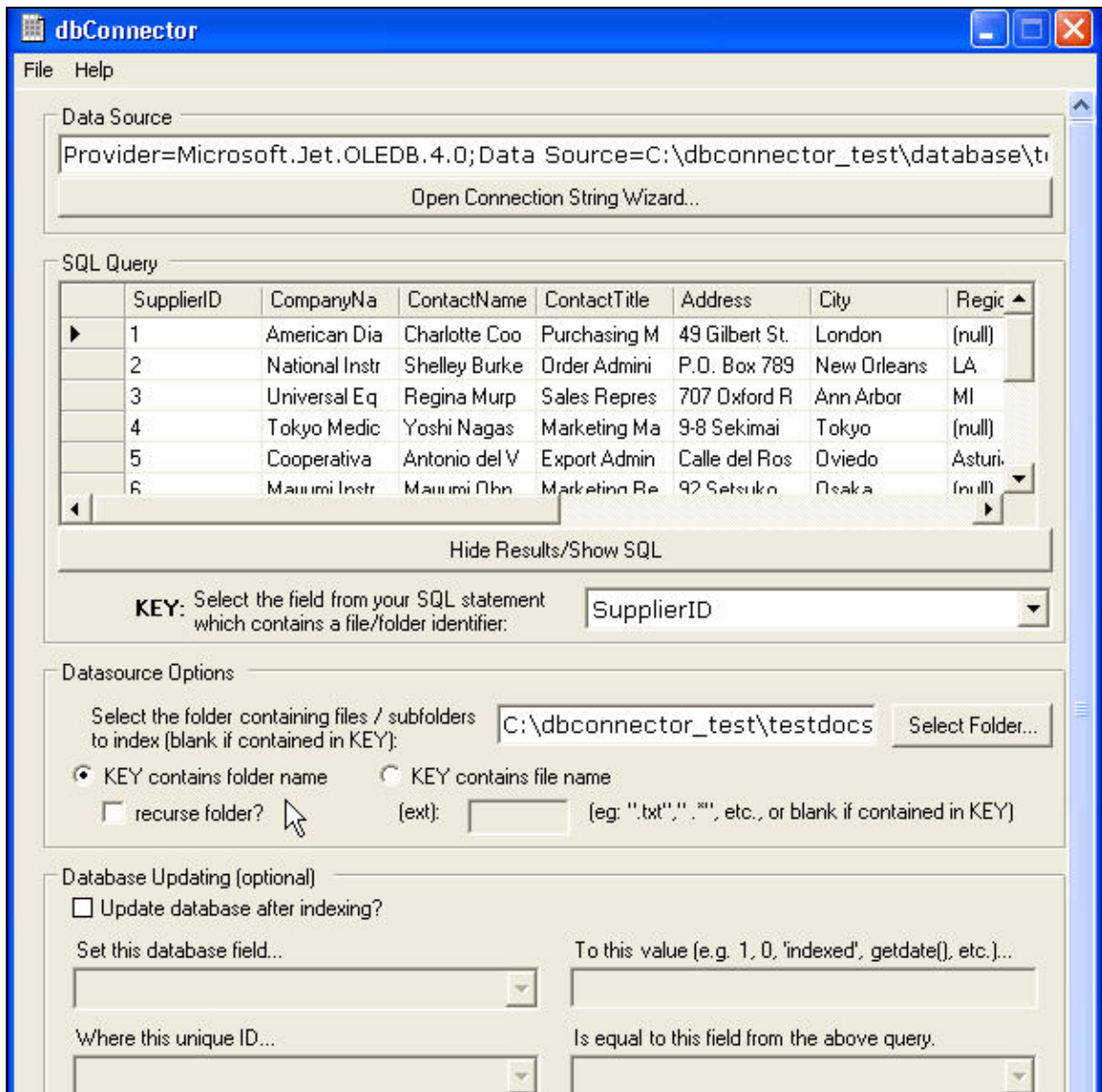


Figure 7

In the Datasource Options box, choose the button next to “KEY contains folder name” and uncheck the “recurse folder?” box.

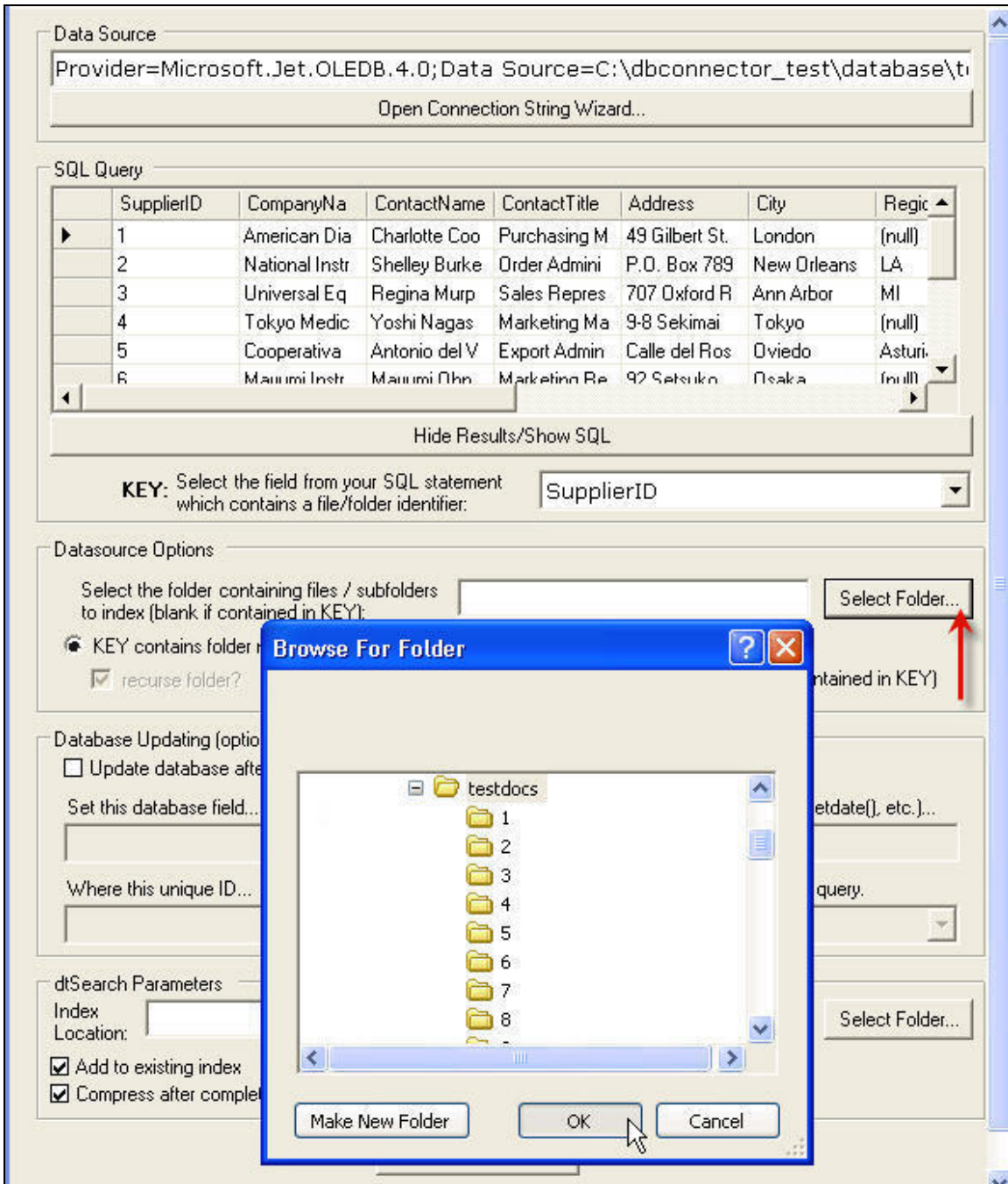


Figure 8

In 'Datasource Options' choose the "Select Folder..." button. You must now select the location of the files that you want to include in your search. Our test files are located in c:\dbconnector_test\testdocs. Browse for and select that folder. (c:\dbconnector_test\testdocs) Click OK.

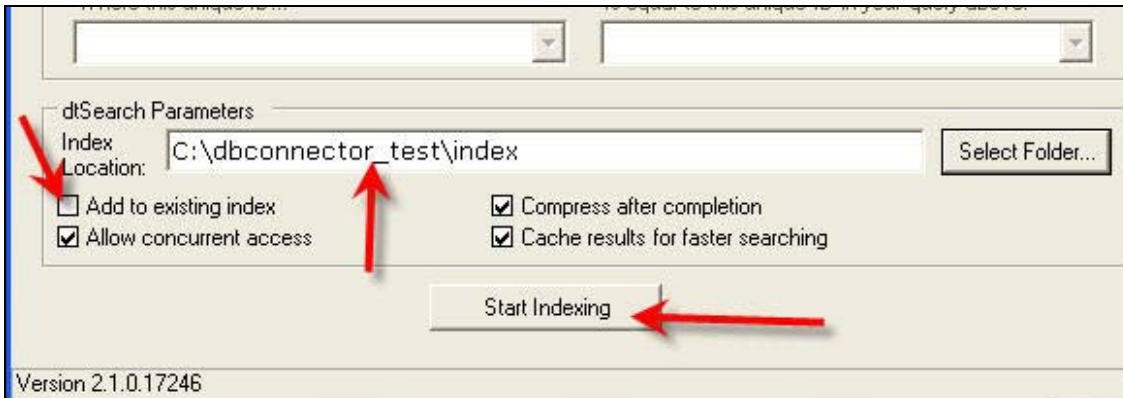


Figure 9

In the dtSearch Parameters section, click on “Select Folder...” and browse for the folder where we would like our index to reside. We are going to place it in C:\dbconnector_test\index.

Choose your desired settings for the four optional checkboxes in dtSearch Parameters. Click “Start Indexing”. (Figure 9)

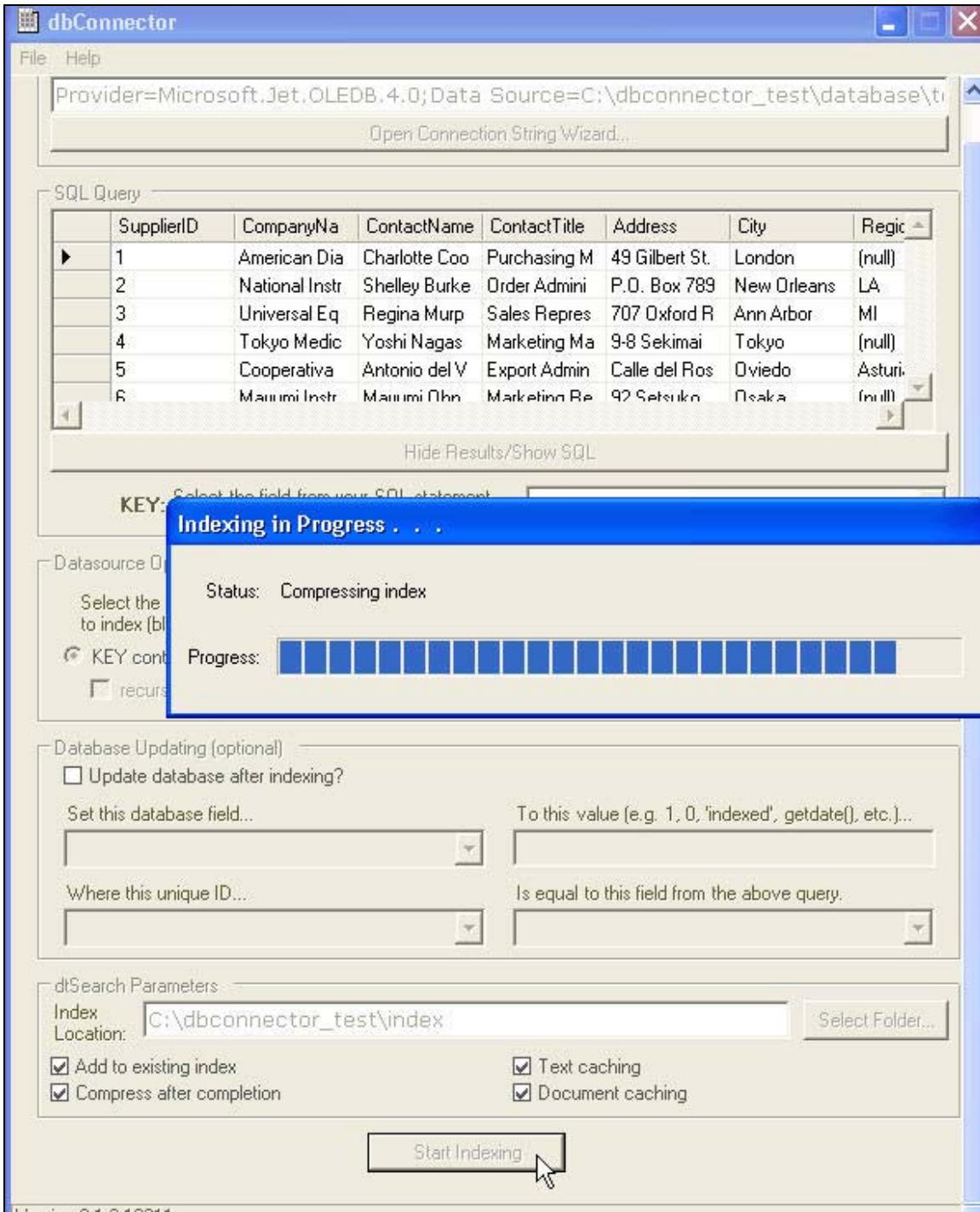


Figure 10

Your index is now being created. When the “indexing complete” box appears, click ‘OK’.

To test your index open dtSearch Desktop. From the toolbar, Select 'Index', 'Index Manager', 'Recognize' the Index you just created, and then use dtSearch Desktop to do a few searches. To carry our test scenario through, you could do a search based on one of the custom fields we included in the index. Supplier 'American Diagnostics' has a contact named 'Charlotte Cooper'. So if we do a search with the following string:

contactname contains cooper

we'll retrieve all documents for Supplier 1, American Diagnostics where Cooper is the contact. See dtSearch results example below.

The screenshot shows the dtSearch Desktop interface. The title bar reads: "contactname contains cooper" -- WAEartipFinal_SM2394.pdf - dtSearch. The menu bar includes File, Edit, Search, Index, View, Options, and Help. The toolbar contains icons for file operations and search. Below the toolbar is a table of search results:

	CompanyName	ContactName	Filename	SupplierID
1	American Diagnostics	Charlotte Cooper	C:\dbconnector_test\testdocs\1\WAEartipFinal_SM2394.pdf	
2	American Diagnostics	Charlotte Cooper	C:\dbconnector_test\testdocs\1\Sm1856.pdf	
3	American Diagnostics	Charlotte Cooper	C:\dbconnector_test\testdocs\1\S107022.pdf	
4	American Diagnostics	Charlotte Cooper	C:\dbconnector_test\testdocs\1\S107018.pdf	
5	American Diagnostics	Charlotte Cooper	C:\dbconnector_test\testdocs\1\200066VV.pdf	
6	American Diagnostics	Charlotte Cooper	C:\dbconnector_test\testdocs\1\OT3-33-03e.pdf	1

Below the table is a toolbar with icons for 'Save a Copy', 'Search', 'Select', and zoom controls (50%). The main content area displays a document page titled "Ear Temperatures Taken with the Braun ThermoScan® Thermometer". The page includes instructions for accurate readings and a list of factors that may interfere with accurate readings.

For Accurate Ear Temperatures, observe the following:

1. For children under 1 year, pull ear straight back. For all other ages, pull ear up and back. This will straighten the ear canal and give the thermometer a clear view of the tympanic membrane.
2. Apply probe cover. Use ONLY Braun ThermoScan probe covers.
3. After placing probe tip in ear, hold "activation" key down until you hear a beep.
4. Keep probe lens clean. Dirt and ear wax on the lens will affect the accuracy of readings. To clean, wipe with alcohol swab and let dry for 5 minutes.
5. Press "mem" key to recall last temperature.
6. Press and hold "mem" key, press "activation" key twice to convert degrees F to degrees C.

The following may interfere with accurate reading:

- ◆ Ambient temperature
- ◆ Dirty or chipped lens
- ◆ Lying on pillow

The page also features an image of the Braun ThermoScan thermometer and a diagram of an ear with the probe tip inserted into the ear canal, pointing towards the tympanic membrane.

Figure 11

Scenario Two

The second scenario provided assumes that you have a collection documents related to records in a database. Our objective is to create an index of documents, and, very importantly, include custom fields from the database in the index. In this scenario the references to the documents are contained in a database table.

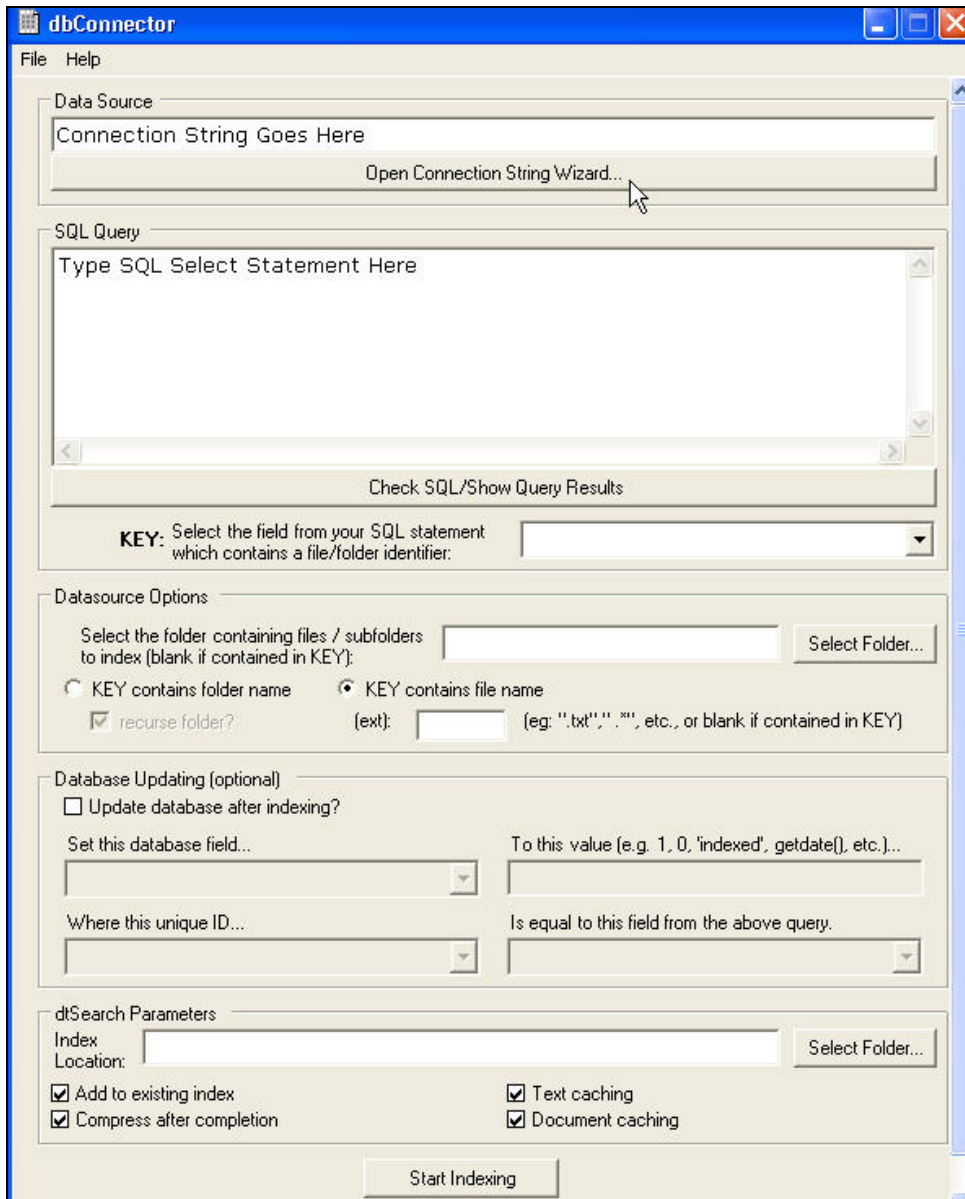


Figure 12

Once you start dbConnector (from the Windows Start Menu / Programs / Cybergroup / dbConnector) the first screen you'll see is shown in Figure 12. Press "Open Connection String Wizard".

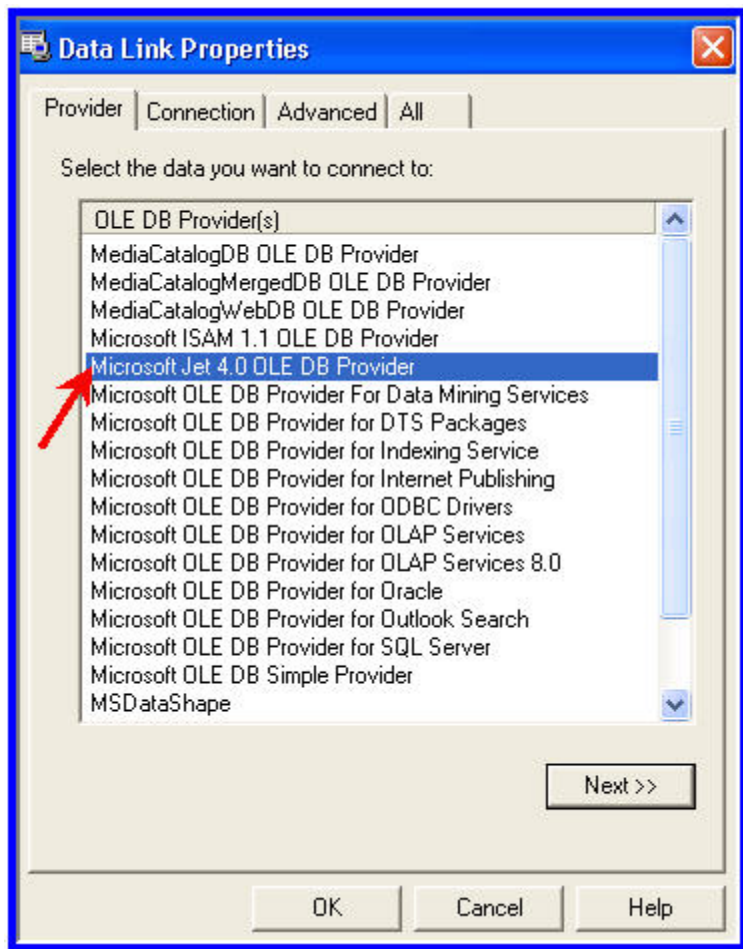


Figure 13

You will now be shown a list of possible OLE DB Providers that you may use to connect dbConnector to your database (Figure 13). For the purposes of the test scenario select 'Microsoft Jet 4.0 OLE DB Provider' and click the 'Next>>' button.

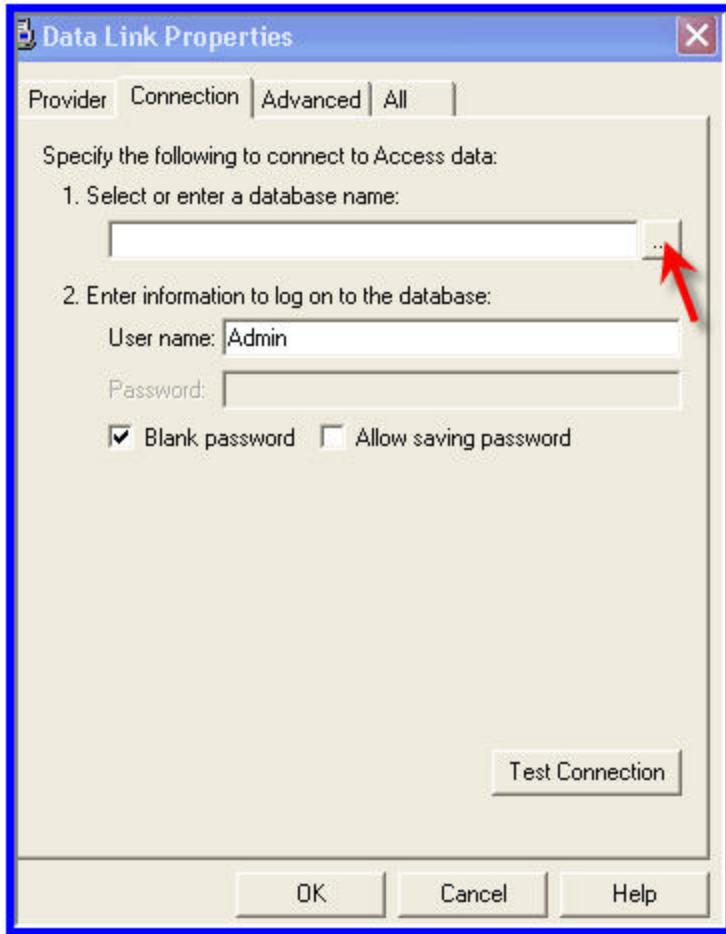


Figure 14

The window shown in Figure 14 will now appear. Press the “...” button to browse for our test database. The database location should be C:/dbConnector_test/database. The filename is testdb.mdb. Find the file and click “Open”.

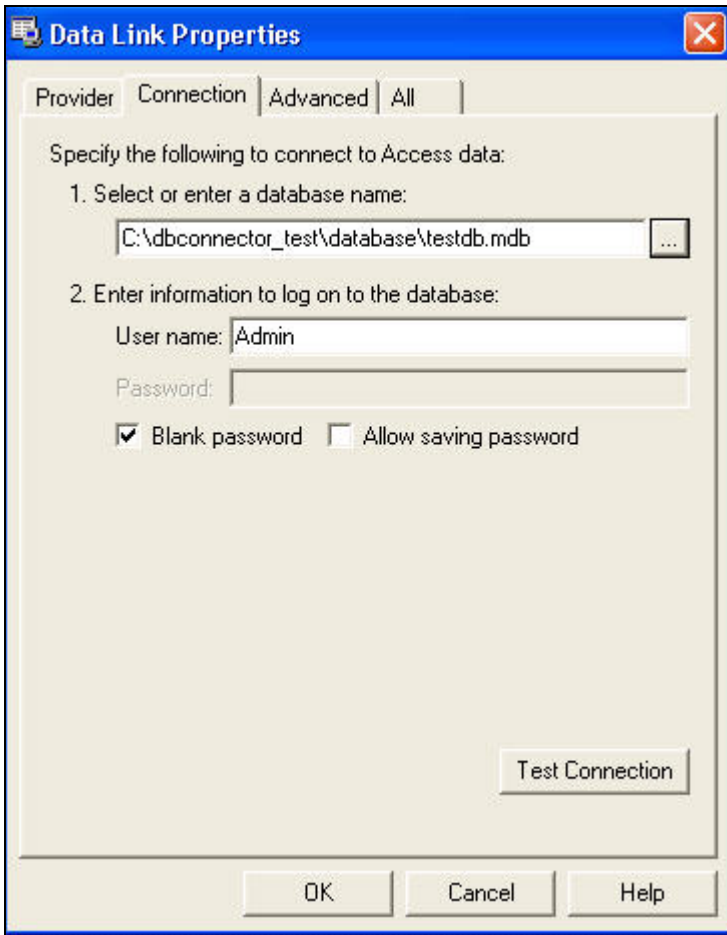


Figure 15

You will then see the screen show in Figure 15. Click "OK".

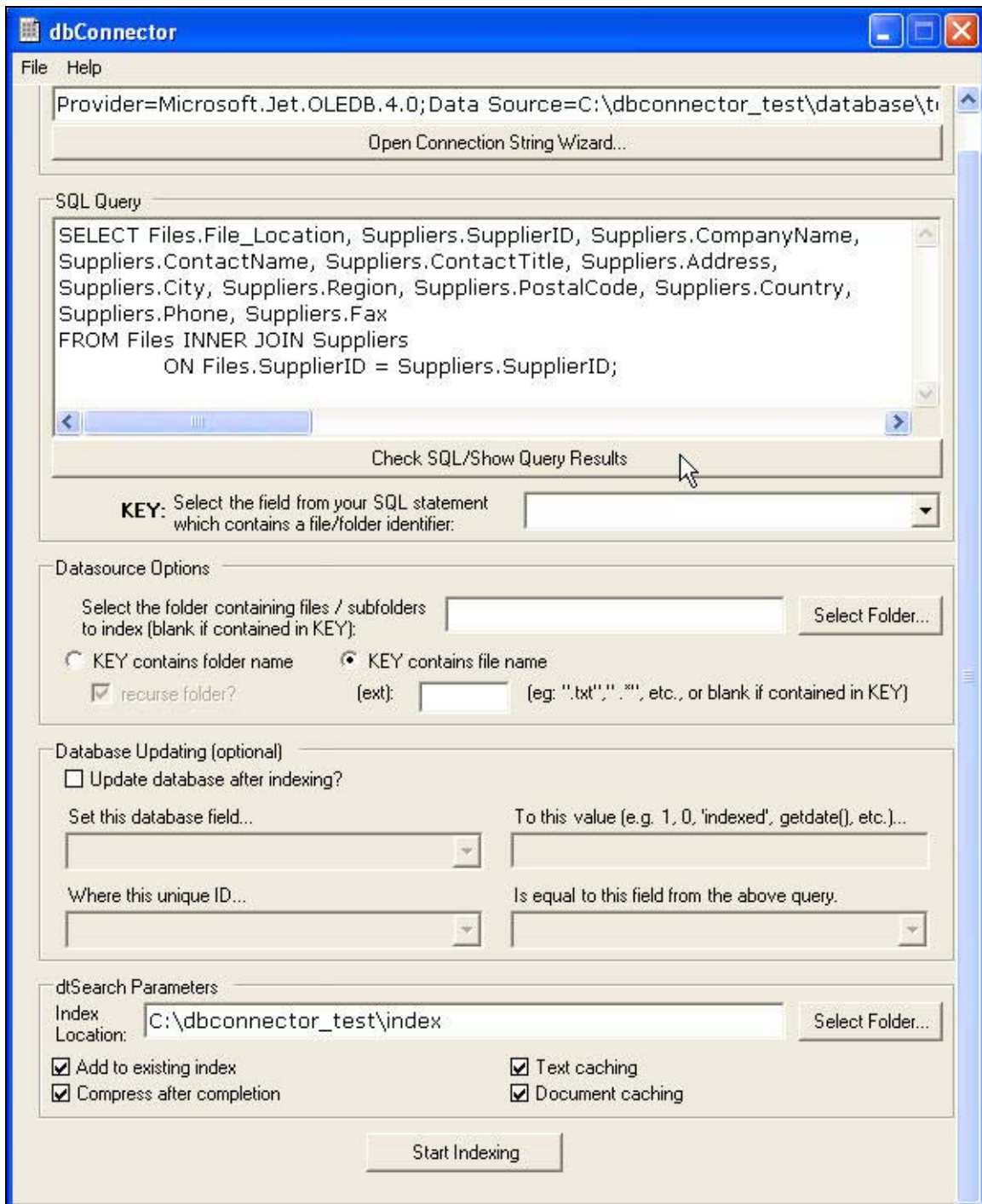


Figure 16
 Enter your SQL statement in the SQL query window. Click the 'Check SQL/Show Query Results' bar. In this example we have joined a table named 'Files' (which contains our documents and a supplier id) with a table named 'Suppliers' and then used the Access Query Builder to create our SQL statement.

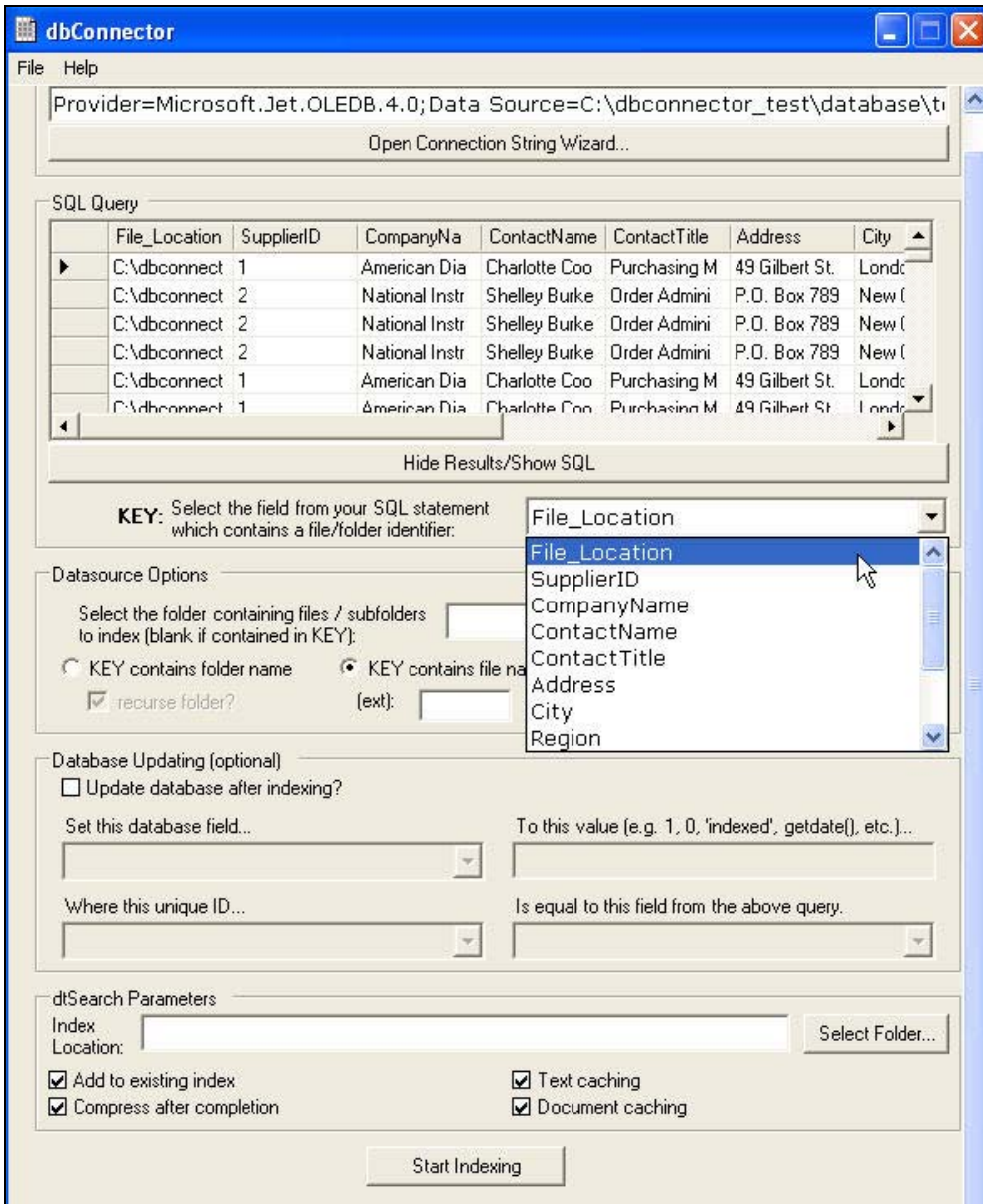


Figure 17

Now, in the KEY dropdown list choose File_Location, as this is where each document's file path is located.

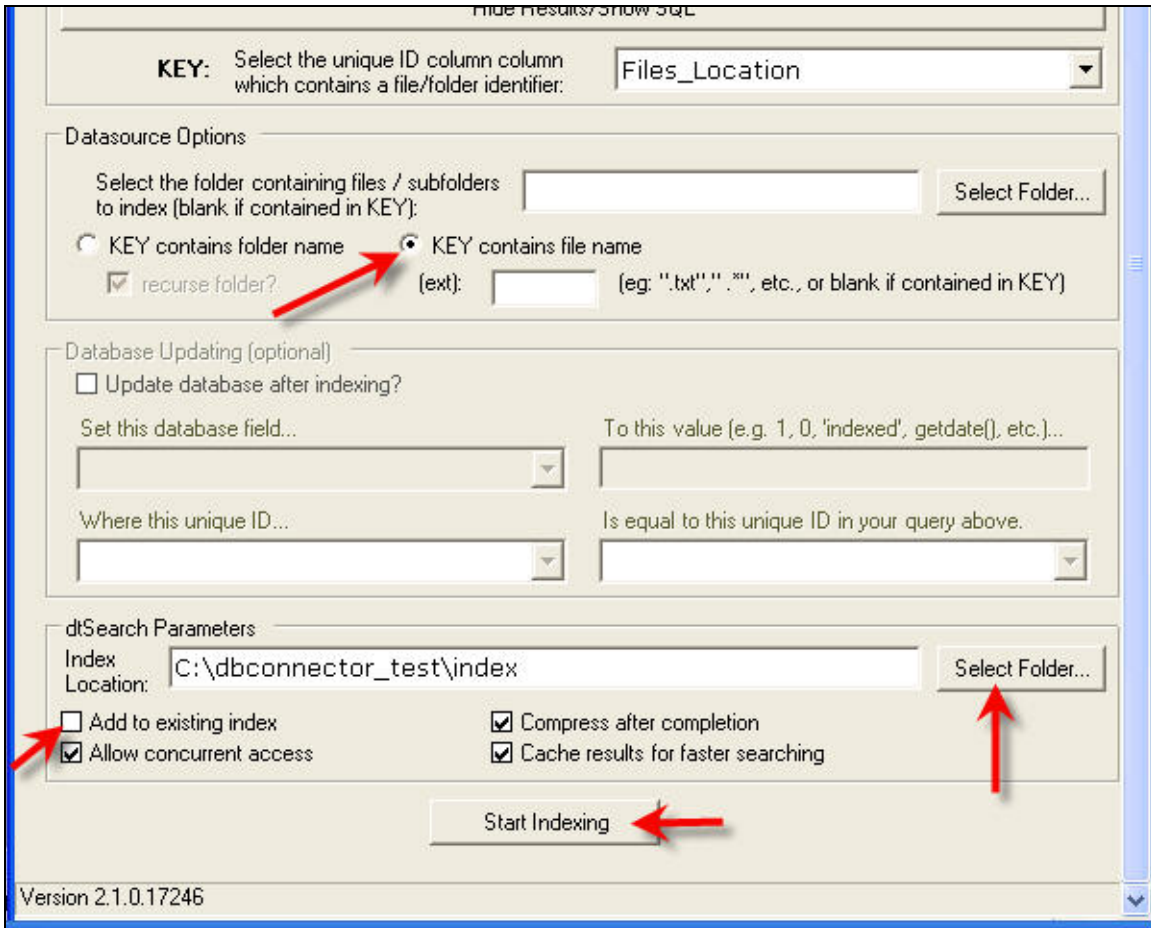


Figure 18

In the Datasource Options section, leave the “Select folder containing files/subfolders...” window blank and choose the radio button “KEY contains file name”.

Next, in the dtSearch Parameters area browse for the folder where you would like to place the index. The path should be C:\dbconnector_test\index.

Check or uncheck the four options in dtSearch parameters based on your desired results. Click “Start Indexing”. (Figure 18)

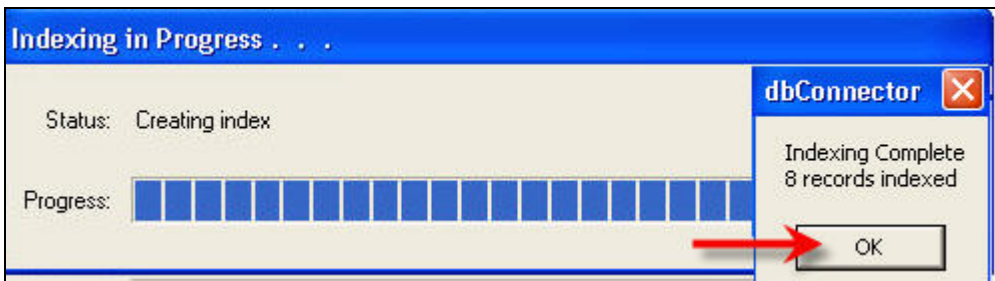


Figure 19

During Indexing you’ll see a progress bar (Figure 19). When the “Indexing Complete” box pops up, click “OK”.

Testing Your Index

To test your index open dtSearch Indexer, click 'Recognize index...' and choose the index you just created in C:\dbconnector_test\index, and then use dtSearch Desktop to do a few searches just as you did as in the first scenario.

For example a search of "Molecular Imaging" will search through all of the files in your table and join the files with supplier information from the database, therefore showing you all documents referencing "molecular imaging" and supplying you with contact and company information associated with that document. See below. (Figure 20)

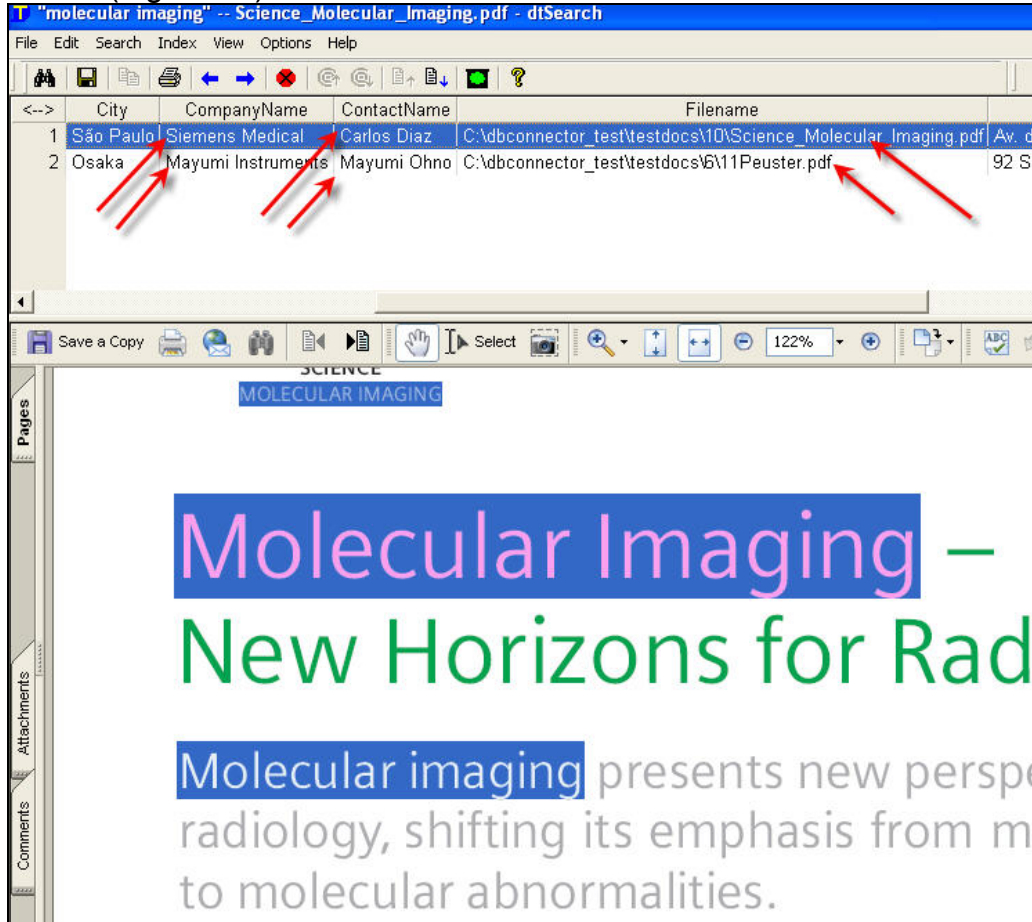


Figure 20

Cybergroup, Inc.
support@cybergroup.com
410-455-5680